Analysing the Performance of Generative Models Trained in a Federated Manner

Exploring the Impact of GANs and Variational Auto-Encoders on Decentralized Data

Alexandru-Nicolae Ojica[1]

Supervisor(s): David Tax, Swier Garst[1]

[1]**EEMCS, Delft University of Technology, The Netherlands**

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 23, 2024

Name of the student: Alexandru-Nicolae Ojica
Final project course: CSE3000 Research Project
Thesis committee: David Tax, Swier Garst, Alex Voulimeneas

Abstract

Federated learning (FL) is an innovative approach in machine learning that enables model training across multiple decentralized devices or servers without sharing local data, thus preserving privacy and utilizing decentralized data. However, a significant challenge in FL is handling non-IID (Non-Identical and Independently Distributed) data, which can adversely affect performance. This paper investigates the impact of federated learning on the performance of various generative models, including Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs), specifically in the context of image and tabular data generation tasks. Our study aims to determine how these generative models perform when trained in a federated manner compared to centralized training. We evaluate the models using several metrics, including classifier accuracy on generated images, Earth Mover's Distance (EMD) for distribution comparison, resemblance, discriminability, downstream utility, and privacy metrics for tabular data. Experiments conducted on the MNIST and CIFAR-10 datasets for image generation, and the Adult and Abalone datasets for tabular data generation, reveal that VAEs exhibit robust and consistent performance across federated and centralized setups. In contrast, GANs show significant performance degradation under federated non-IID conditions. The results indicate that VAEs can effectively address the non-IID data challenge in FL by generating high-quality synthetic data, thereby enhancing model generalizability and stability. The framework used for executing the experiments in this study can be found at https://github.com/alexojica/research-project-experiments.

1 Introduction

In recent years, the development of advanced generative models such as Generative Adversarial Networks (GANs)[5] and Variational Auto-Encoders(VAEs)[13] has revolutionized the field of machine learning by enabling the creation of high-quality synthetic data. These models have found applications in various domains, including image generation, data augmentation[28], and anomaly detection[12]. However, traditional training of these models requires centralized datasets, which can pose significant privacy and security concerns. This has paved the way for the emergence of Federated Learning (FL)[16], a paradigm that aims to train machine learning models using data distributed across multiple locations without requiring data to be centrally stored.

Federated Learning (FL) facilitates the training of machine learning models by leveraging data that remains localized on multiple devices or servers. This approach addresses privacy concerns by ensuring that raw data never leaves its original location. Instead, models are trained locally and only model updates are shared and aggregated to create a global model [16]. Despite its advantages, FL faces substantial challenges, particularly when dealing with non-Identical and Independently Distributed (non-IID) data. Non-IID data can significantly degrade model performance because the data distribution varies across different nodes, leading to biased and less generalizable global models [30].

One promising approach to mitigate the effects of non-IID data in FL is local data augmentation across nodes[29]. This technique involves training a generative model in a federated manner and using it to augment the local datasets with synthetic data. However, further research into the effects of federated learning on generative models is necessary to better understand and advance this method as it directly impacts the quality and diversity of the data generated for augmentation. This is essential for producing augmented data that accurately reflects the heterogeneity of the real-world data, thus improving model robustness and performance.

This research seeks to investigate the following question: **How does the performance of different generative models (GANs and Variational Auto-Encoders) get affected by training in a federated manner?**

This study employs an empirical research design to answer the aforementioned question. By addressing both image generation and tabular data generation tasks, the study aims to explore the feasibility and efficacy of using generative models to handle the challenges posed by non-IID data in federated learning.

The structure of this report is as follows: first, we provide a detailed background on generative models and federated learning, highlighting their individual advancements and challenges. Next, we present our methodology for evaluating the performance of GANs and VAEs in a federated setup. We then discuss the experimental results, analyzing the impact of federated training on these generative models. Finally, we conclude with a summary of our findings and suggest directions for future research.

2 Background

2.1 Generative Models

Among the most prominent generative models are Generative Adversarial Networks (GANs) and Variational Auto-Encoders (VAEs). Each of these models has unique characteristics and applications.

Generative Adversarial Networks (GANs)

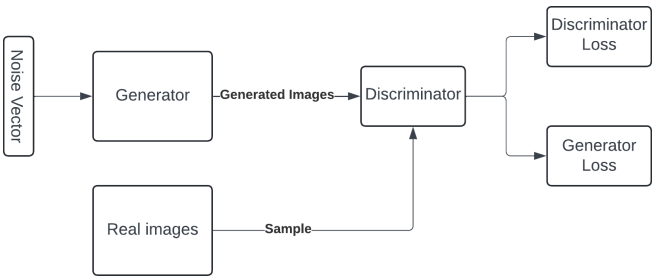


Figure 1: Architecture of a Generative Adversarial Network

Generative Adversarial Networks (GANs) were introduced by Ian Goodfellow and his colleagues in 2014 [5]. A GAN consists of two neural networks: a generator and a discriminator (Figure 1). The generator creates synthetic data samples, while the discriminator evaluates them against real data samples. The two networks are trained simultaneously in a zero-sum game framework, where the generator aims to produce data indistinguishable from real data, and the discriminator strives to distinguish between real and synthetic data. This adversarial process continues until the generator produces highly realistic data.

Variational Auto-Encoders (VAEs)

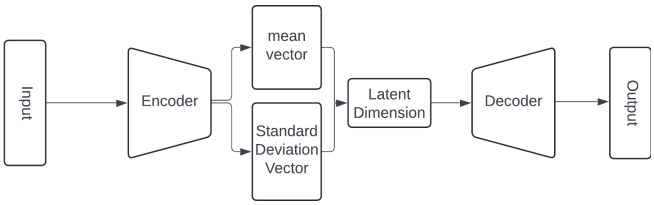


Figure 2: Architecture of a Variational Auto-Encoder

Variational Auto-Encoders (VAEs), introduced by Kingma and Welling in 2013 [13], are another popular class of generative models. VAEs are designed to learn the underlying distribution of the data by encoding input data into a latent space and then decoding it back to the original space (Figure 2). Unlike traditional auto-encoders, VAEs introduce a probabilistic approach by modeling the latent space as a distribution rather than a fixed vector. This allows VAEs to generate new data samples by sampling from the learned latent distribution.

2.2 Federated Averaging (FedAvg)

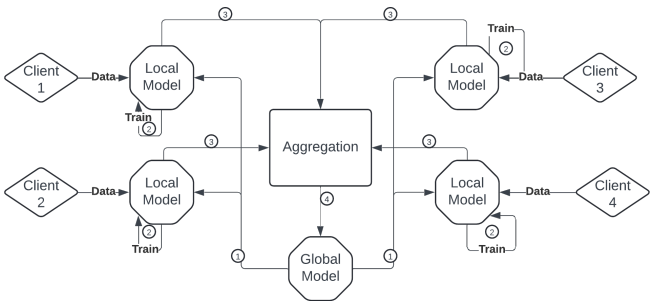


Figure 3: FedAvg Algorithm

Federated Averaging (FedAvg) is one of the most widely used algorithms in federated learning, introduced by McMahan et al. in 2017 [16]. FedAvg works by iteratively training local models on individual devices and then averaging the model weights to obtain a global model. The algorithm follows these steps:

1. **Initialization:** The global model is initialized and sent to all participating devices.
2. **Local Training:** Each device trains the model on its local data for a set number of epochs.
3. **Model Update:** The locally trained models are sent back to a central server.
4. **Averaging:** The central server aggregates the model weights by computing their average, resulting in an updated global model.
5. **Iteration:** Steps 2-4 are repeated until the model converges.

FedAvg is particularly effective in scenarios where data is distributed across numerous devices, such as smartphones or IoT devices, and helps preserve data privacy since raw data never leaves the local devices. Despite its advantages, FedAvg faces challenges when dealing with non-IID data distributions, which can lead to biased and suboptimal global models.

3 Methodology and Experimental Setup

All algorithms are implemented in Python 3.9, making use of the PyTorch[21] framework.

3.1 Federated Learning Setup

The federated learning framework utilized in this study is a local simulation of the Federated Averaging (FedAvg) algorithm. This framework splits the data between 100 users and, at each step, selects ten random users. A deepcopy of the central model is trained with each selected user's data for one local epoch, and then the weights from each trained model are averaged. This process is repeated for a specified number of global epochs, depending on the model.

3.2 Data and Data Preparation

The experiments utilize several datasets for various data generation tasks. For image generation, we use the MNIST[3] and CIFAR-10[15] datasets. The MNIST dataset, consisting of black and white images of handwritten digits, was selected for its simplicity, allowing less powerful models such as VAEs to produce meaningful results. In contrast, the CIFAR-10 dataset contains RGB images of 10 different classes, presenting a more complex and challenging scenario suitable for testing the capabilities of more advanced models like GANs.

For tabular data generation tasks, we employ the Adult[1], also known as the "Census Income" dataset, and the Abalone[18] datasets. The Adult dataset, encompassing data on approximately 48,000 individuals including attributes like age, work class, education, and capital gains, was chosen due to its large size and the presence of both categorical and numerical columns. The Abalone dataset, containing measurements such as length, diameter, and height of marine snails (abalone), was selected for its smaller size and the predominance of numerical columns, making it ideal for evaluating regression tasks.

Normalization transformations are applied to the MNIST and CIFAR-10 datasets during preprocessing to ensure consistency and enhance model training efficiency. Specifically, the MNIST dataset undergoes normalization to scale its pixel values to a range centered around zero with a standard deviation of 0.5. For the CIFAR-10 dataset, normalization is

applied to adjust each color channel (red, green, and blue) independently to have a mean of 0.5 and a standard deviation of 0.5. For the tabular datasets no specific preprocessing was performed.

3.3 Data Distribution

The experiments are divided into three data distribution setups to evaluate different scenarios:

- Centralized: All data is available in a single location for model training.
- Federated-IID
- Federated-Non-IID

To simulate Federated-IID and Federated-Non-IID data distributions, we follow the approach introduced by McMahan et al. [16]. For the IID setup, the data is shuffled and then evenly partitioned among the clients. For the Non-IID setup, the data is first sorted by class or attribute, divided into shards, and then assigned to clients such that each client receives only a subset of the classes or attributes. For tabular data that contains more than one categorical column such as the Adult dataset, the attribute after which the data is sorted is a categorical column chosen at random.

3.4 Model Architectures Used

In our research, we employed different generative models tailored to the specific characteristics of the datasets used. For the MNIST dataset, we utilized a simple Conditional GAN (CGAN) [17] and a Conditional VAE (CVAE) [6]. These models were chosen due to their demonstrated effectiveness in generating high-quality synthetic data for simpler image datasets, making them well-suited for the relatively low-complexity MNIST dataset.

For the CIFAR-10 dataset, which is more complex and contains higher-dimensional images, we required a more robust architecture. Thus, we implemented a modified Auxiliary Classifier GAN (ACGAN) [19] featuring convolutional layers instead of linear layers, as described in the Deep Convolutional GAN (DCGAN) paper [23]. This modification leverages the power of convolutional networks to better capture the intricate patterns present in CIFAR-10 images.

To evaluate the performance of VAEs on the CIFAR-10 dataset, we employed a simple Conditional VAE (CVAE) with convolutional layers, replacing the traditional linear layers. This adjustment allows the model to better handle the complexity of the image data by utilizing the spatial hierarchies learned through convolutional layers.

For tabular data, we used the CTGAN and TVAE models as introduced by Lei Xu et al. in their paper "Modeling Tabular Data Using Conditional GAN" [27]. These models were selected due to their state-of-the-art performance on various tabular datasets, demonstrating their capability to effectively model the dependencies and structures inherent in such data.

3.5 Image Generation Evaluation Metrics

For image generation tasks, the following metrics are used:

- **Classification Score**: The accuracy of a strong classifier trained on real images when used on generated images. For MNIST, a simple CNN(Convolutional Neural Network)[20] with 2 convolutional layers, having 99.6% accuracy, is used, while for CIFAR-10, a ResNet-50[8] model with 85% accuracy is employed.
- **Earth Mover's Distance (EMD)**: This metric measures the similarity between the distributions of real and generated images.[24]

3.6 Tabular Data Metrics

The metrics chosen for evaluating the performance of tabular data generators are Resemblance, Downstream Utility, Privacy and Discriminability. The first three were proposed by Hernandez et al. in 2023[9], while the latter was introduced by Goodfellow et al. in 2014[5]. All metrics used in evaluated the tabular data are normalized to take values from 0 to 100 in order to make the comparison between models easier.

Resemblance Score

The Resemblance score ($\theta_{\text{resemblance}}$) evaluates how closely the synthetic data distribution matches the real data distribution. It achieves this by averaging the correlation score ($\rho_{\text{correlation}}$) with the complements of the Jensen-Shannon [4] and Kolmogorov-Smirnov [14] distances between the two datasets. The correlation score ($\rho_{\text{correlation}}$) is determined using Theil's U [26] for correlations between categorical columns and Pearson's r [22] for numerical columns.

$$\theta_{\text{resemblance}} = 100 \times \min\left(1, \max\left(0, \mu_c\right)\right)$$

$$\mu_c = \frac{1}{3}\left(\rho_{\text{correlation}} + \left(1 - D_{\text{Jensen-Shannon}}\right) + \left(1 - D_{\text{Kolmogorov-Smirnov}}\right)\right)$$

$$\rho_{\text{correlation}} = \frac{1}{|\mathbf{C}|}\sum_{c \in \mathbf{C}} \rho_c,$$

$$\text{where } \rho_c = \begin{cases} \text{Theil's U}(\mathbf{R}[c], \mathbf{S}[c]) & \text{if } c \in \mathbf{cat} \\ \text{Pearson's } r(\mathbf{R}[c], \mathbf{S}[c]) & \text{if } c \notin \mathbf{cat} \end{cases} \text{where}$$

$\mathbf{R}$, $\mathbf{S}$, $\mathbf{cat}$ and $\mathbf{C}$ stand for the Real dataset, Synthetic dataset, the set of categorical columns and the set of all columns, respectively.

Discriminability Score

The discriminability score $\theta_{\text{discriminability}}$ evaluates how hard it is for a classifier to discriminate between synthetic and real data. The higher the error of the discriminator, the better the synthetic data is considered as it "fools" the classifier.

$$\theta_{\text{discriminability}} = 100 \times (1 - \text{PMSE})$$

where PMSE is the Predictive Mean Squared Error of an XGBoost[2] classifier trained to classify data points into either real or synthetic. The PMSE is calculated by evaluating the performance of the classifier on a test set consisting of both real and synthetic data points.

Downstream Utility Score

The downstream utility score, denoted as $\theta_{\text{downstream\_utility}}$, evaluates the performance of synthetic data in comparison to real data when used in typical machine learning tasks. This is achieved by training different models on the real and synthetic datasets and then evaluating them on real data to see

how they compare. This test evaluates the quality of the synthetic data for both classification and regression by training classifiers to predict the categorical features of the data and regressors to predict the numerical features of the data given an incomplete record. The final score averages the utilities for the classification and regression tasks.

$$\theta_{\text{downstream\_utility}} = \frac{1}{2}\left(\theta_{\text{classification}} + \theta_{\text{regression}}\right)$$

For classification tasks, the utility score is based on the macro F1 score:

$$\theta_{\text{classification}} = 100 \times \frac{\text{Macro F1 (real)} - \text{Macro F1 (synthetic)}}{\text{Macro F1 (real)}}$$

For regression tasks, the utility score is based on the D^2 absolute error:

$$\theta_{\text{regression}} = 100 \times \frac{\text{D}^2\text{(real)} - \text{D}^2\text{(synthetic)}}{\text{D}^2\text{(real)}}$$

Here, the models used are `XGBoostClassifier`[2] for classification tasks and `XGBoostRegressor`[2] for regression tasks. The use of `XGBoost` models is motivated by their robustness and efficiency:

- **XGBoostClassifier:** Known for its high performance in classification tasks, XGBoost provides a powerful and scalable algorithm that handles various types of data effectively.
- **XGBoostRegressor:** This model is well-regarded for its predictive accuracy in regression tasks and its ability to handle large datasets with ease.

The choice of the macro F1 score[25] for classification tasks and D^2 absolute error[7] for regression tasks is motivated by the need to evaluate different aspects of model performance:

- **Macro F1 Score:** This score is particularly useful for classification tasks as it accounts for both precision and recall, and it is particularly effective in scenarios with class imbalance. By considering the macro F1 score, we ensure that the classifier performs well across all classes, not just the majority class.
- **D^2 Absolute Error(D^2 in the above formula):** This metric is chosen for regression tasks as it measures the accuracy of the model's predictions by calculating the squared differences between the predicted and actual values. It provides a clear indication of the model's predictive accuracy.

Privacy Metric: Attribute Inference Attack Score

The privacy metric is calculated by performing an Attribute Inference Attack(AIA)[9]. In this scenario the attacker has access to the publicly available synthetic dataset and a percentage of columns from the real dataset, considered compromised. Then for a subset of columns from the real dataset that the attacker does not have access to, considered sensitive columns, a model is trained on the synthetic data to predict these values given the compromised columns. In the experiments performed as part of this study, 50% of the columns are considered compromised and 25% are considered sensitive. An XGBoostClassifier is used for the categorical columns and an XGBootRegressor is used for the numerical columns. The Synthetic and Real scores are represented by the macro f1 score for the categorical attributes and the d2 absolute error for the numerical attributes when evaluated on real data.

The AIA score is then computed as follows:

$$\text{AIA Score} = 1 - \frac{\text{quantile}_{0.9}(\text{Synthetic Scores}) - 0.5}{\text{quantile}_{0.9}(\text{Real Scores}) - 0.5}$$

The score is clipped to the range [0, 1]:

$$\text{AIA Score}_{\text{clipped}} = \min(1, \max(0, \text{AIA Score}))$$

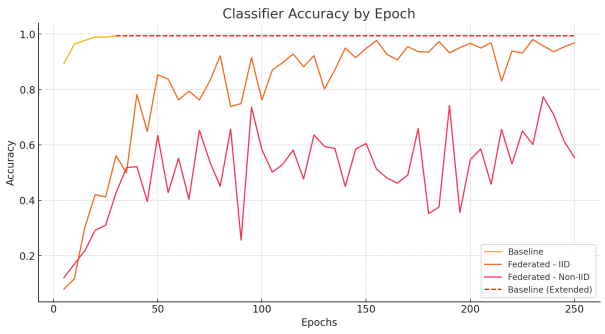
Finally, the clipped AIA score is scaled by 100 and rounded to the nearest integer:

$$\text{Final AIA Score} = 100 \times \text{AIA Score}_{\text{clipped}}$$

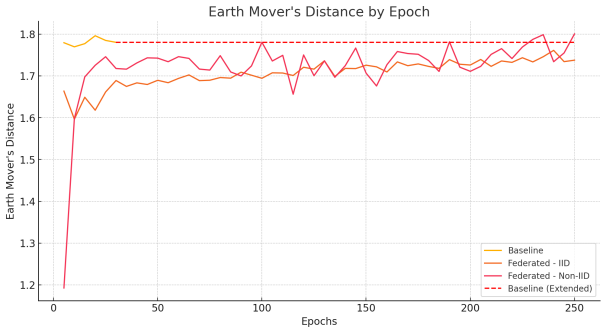
4 Results

The following section describes the experiments performed in order to answer the research question.

4.1 Experiments on Image Generation Tasks

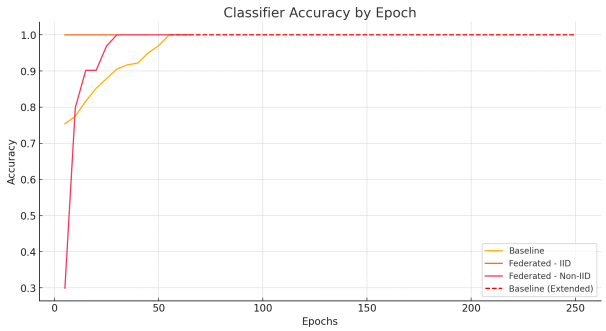


(a) Classification Score vs. Epochs for Conditional GAN on MNIST

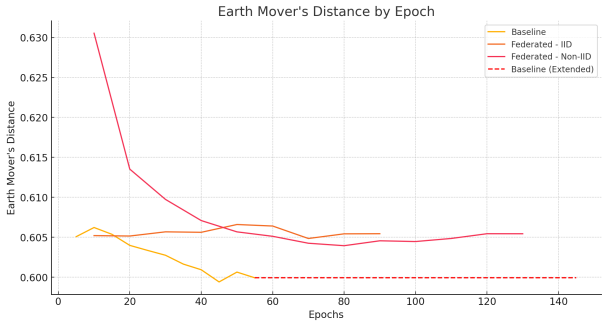


(b) EMD vs. Epochs for Conditional GAN on MNIST

Figure 4: Performance Metrics for Conditional GAN on MNIST



(a) Classification Score vs Epochs for Conditional VAE on MNIST



(b) EMD vs Epochs for Conditional VAE on MNIST

Figure 5: Performance Metrics for Conditional VAE on MNIST

Experiments on MNIST

As can be seen in Figure 4a the Conditional GAN the baseline achieves almost perfect accuracy after as little as 30 epochs, while for the federated-IID setup it starts to converge after 150 epochs, never reaching the accuracy where the baseline converged. For the federated-non-IID setup the model does not reach a stable state even after 250 epochs, experiencing high variance in its performance and never reaching the performance achieved by the other 2 setups. On the other side, as Figure 5a shows, the Conditional VAE converges in all 3 setups relatively fast, achieving perfect accuracy in the federated-IID setup after as little as five epochs. The federated-non-IID setup converged a bit slower, after about 30 epochs, still beating the baseline which achieved perfect accuracy after more than 50 epochs.

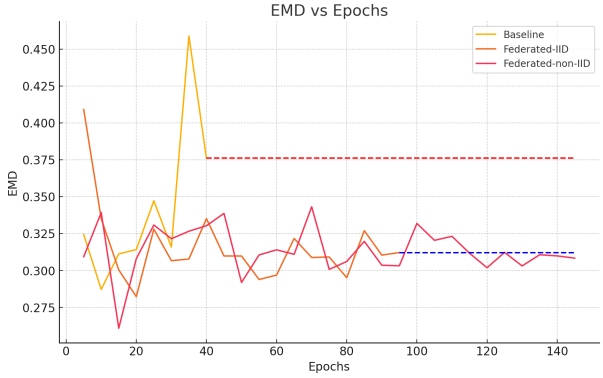
The EMD values for Conditional GAN(Figure 4b) remain relatively high across all setups, deteriorating as the number of epochs increases. This, taking into consideration the convergence of the classifier score, indicates that the GAN might overfit to the characteristics that the classifier used to calculate the classifier score recognises, creating samples that are further from the real images with regards to certain attributes. The Conditional VAE shows an improvement with regards to the EMD values(Figure 5b as the number of epochs increases, indicating that the model improves in recognising all aspects of the data, even in the federated-non-IID setup.

In training the CGAN a learning rate of 0.0002 was used and the training was stopped after 30 epochs for the baseline setup to avoid overfitting. In the federated-IID and non-IID setups the training was stopped after 250 epochs as the models learned slower. For the CVAE a learning rate of 0.001 was used and the training was stopped after 50 epochs for the baseline version, while for the federated-IID and non-IID setups 95 and 135 epochs were used, respectively.

When it comes to the stability of the results, for both the CGAN and CVAE similar values were obtained in different runs. For the CGAN the baseline converges relatively fast, after about 20 to 30 epochs, while the federated-IID model takes about 230-250 rounds to converge. The federated-non-IID model never reaches the accuracy level of the other two, having high fluctuations in the classifier score. The CVAE is even more stable as all three setups converge every run, showing a downward trend in the EMD values, as depicted in figure 4b.

Experiments on CIFAR-10

(a) Classification Score vs Epochs for Conditional GAN on CIFAR-10

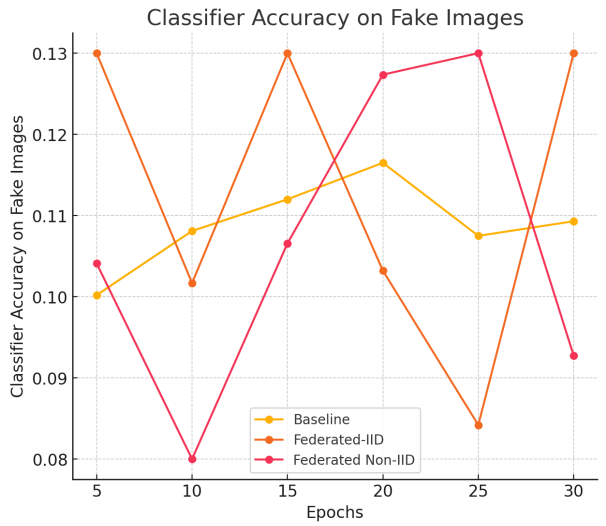


(b) EMD vs Epochs for Conditional GAN on CIFAR-10

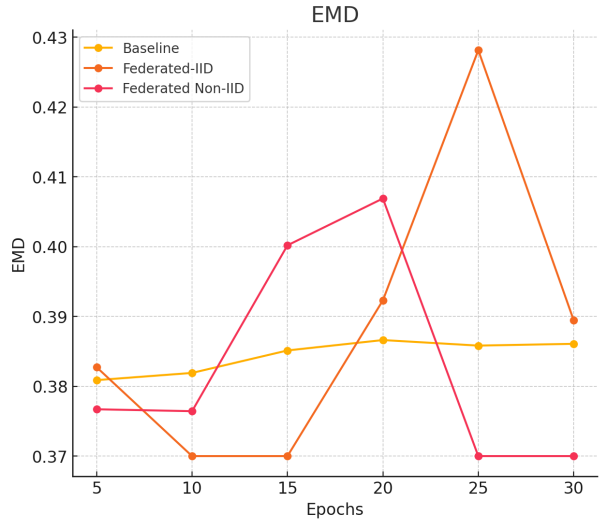
Figure 6: Performance Metrics for Conditional GAN on CIFAR-10

Figure 6a shows the classifier score that the GAN model achieved as the number of epochs grows. The baseline model performed the best, achieving a Classifier Accuracy of 0.54 after about 30 epochs. The federated-IID setup also achieved remarkable results with a 0.35 classifier accuracy after about 90 epochs. The federated-non-IID setup achieved an accuracy of 0.18 after 150 epochs. However, none of the models trained in a federated manner managed to come close to the

performance achieved by the baseline model. Figure 7a illustrates the accuracy achieved by the Conditional VAE model. The model in the baseline setup was more stable than the models trained in a federated manner. In the figure it can also be observed that the CVAE did not perform nearly as well as the GAN, indicating that more powerful models are needed for a dataset as complex as CIFAR-10.



(a) Classification Score vs Epochs for Conditional VAE on CIFAR-10



(b) EMD vs Epochs for Conditional VAE on CIFAR-10

Figure 7: Performance Metrics for Conditional VAE on CIFAR-10

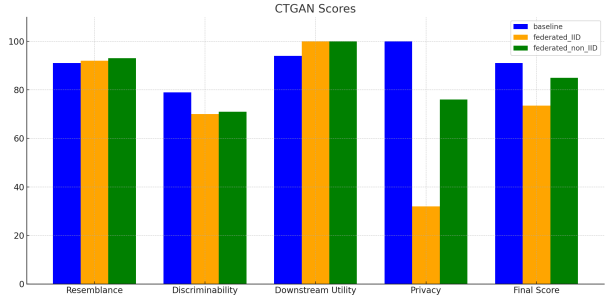
Figure 6b shows the EMD for the GAN models as the number of epochs increases. The baseline model's EMD experiences high fluctuations, deteriorating over time. The federated setups experience less fluctuations, both ending up with an improved EMD value as the training stops. As in the case of the CGAN used for the experiments on MNIST, the Earth Mover's Distance values obtained by the GAN indicate an overfitting on specific features of the data that the classifier recognises.

The EMD values for the CVAE show less variation across the setups(Figure 7b. The centralized setup maintains relatively low EMD values, indicating effective learning of data distributions. Federated setups, both IID and non-IID, show higher but comparable EMD values, highlighting that CVAEs can manage distributional differences to some extent even under non-IID conditions.

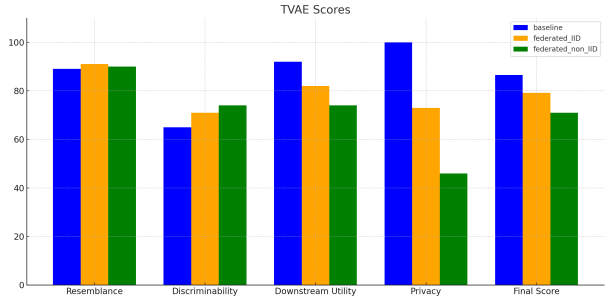
As in the case of experiments on MNIST, the experiments on CIFAR-10 revealed similar results each run, with the GAN performing well in both the baseline and federated-IID settings while never exceeding 0.2 Classifier Accuracy in the federated-non-IID setting. The EMD values showed fluctuations as depicted in Figure 6b across all runs. The same applies to the CVAE model, where the baseline always remained more stable than the federated settings in terms of values for both the Classifier Accuracy and EMD.

4.2 Experiments on Tabular Data Generation Tasks

Experiments on the Adult Dataset



(a) Scores for the CTGAN Generator



(b) Scores for the TVAE Generator

Figure 8: Performance Metrics for CTGAN and TVAE on the Adult dataset

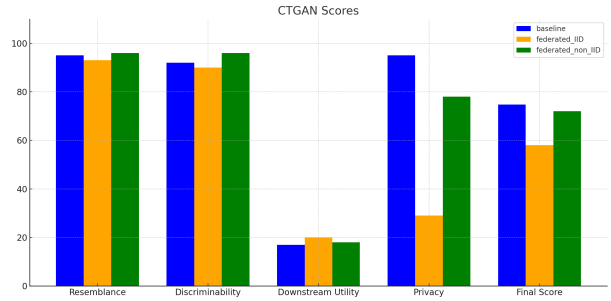
The baseline models for both CTGAN and TVAE generally outperformed their federated counterparts, highlighting the challenges and trade-offs associated with implementing federated learning in synthetic data generation. The CTGAN baseline model demonstrated the strongest overall performance, excelling particularly in Privacy and Downstream Utility. This indicates that the centralized CTGAN model can generate high-quality synthetic data that closely mim-

ics real data distribution while maintaining utility for downstream tasks and preserving privacy.

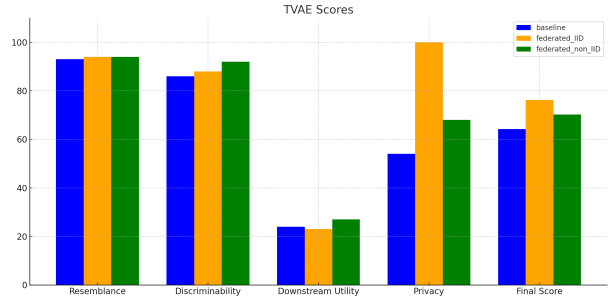
In contrast, the federated learning implementations of CTGAN revealed some drawbacks. The federated IID version of CTGAN showed significant strength in Downstream Utility but suffered a notable drop in Privacy. The federated non-IID configuration of CTGAN achieved a more balanced performance but still did not match the baseline model, indicating a promising direction in solving the non-IIDness problem in federated learning.

The TVAE models followed a similar trend, with the baseline configuration outperforming the federated versions. The federated IID version maintained a relatively balanced performance but did not excel in any specific metric. The federated non-IID version of TVAE showed the lowest overall performance, indicating that non-IID data distribution has a more pronounced negative impact on TVAE than on CTGAN.

Experiments on the Abalone Dataset



(a) Scores for the CTGAN Generator



(b) Scores for the TVAE Generator

Figure 9: Performance Metrics for CTGAN and TVAE on the Abalone dataset

For the Abalone dataset, the federated IID configuration of TVAE demonstrated the strongest overall performance with a final score of 76.2. This indicates that the TVAE model can effectively handle federated IID scenarios, maintaining high scores in Resemblance and Privacy. The CTGAN baseline model follows closely with a final score of 74.8, showing strong performance across all metrics but slightly lower in Discriminability and Downstream Utility compared to the TVAE federated IID model.

The federated non-IID configurations for both CTGAN and TVAE showed lower performance compared to their federated IID counterparts, reflecting the additional challenges posed by non-IID data distribution. The CTGAN federated non-IID model achieved a final score of 72.0, while the TVAE federated non-IID model scored 70.2. These results highlight that while federated learning can be implemented, the non-IID nature of the data can negatively impact the generator's performance, particularly in terms of Privacy and Discriminability.

The baseline configurations for both models showed a decline in performance compared to the Adult dataset. The TVAE baseline model scored 64.2, indicating moderate performance across all metrics but not excelling in any particular area. The CTGAN federated IID configuration had the lowest overall performance with a final score of 58.0, primarily due to its low Privacy and Downstream Utility scores, suggesting that the IID assumption in federated learning may not hold well for the Abalone dataset.

For both datasets, the results were consistent across multiple runs, with a variation of approximately 2 points for each metric.

5 Responsible Research

5.1 Ethical Aspects

The ethical considerations of this research primarily revolve around data privacy and the responsible use of synthetic data. Federated learning inherently enhances privacy by ensuring that raw data remains localized, reducing the risk of data breaches and unauthorized access. However, the generation of synthetic data, particularly when using generative models, must be handled carefully to avoid inadvertently revealing sensitive information.

It is crucial to ensure that synthetic data cannot be reverse-engineered to re-identify individuals, especially in sensitive domains such as healthcare and finance. As seen in the experiments performed, the synthetic data is quite resistant to such attacks. However, there are some cases where this could be improved such as for the federated-IID setup of the CTGAN generator and the federated-non-IID setup of the TVAE generator. Adopting rigorous privacy-preserving techniques, such as differential privacy[10], can mitigate these risks. Moreover, ethical guidelines and compliance with regulations like GDPR must be followed to safeguard data privacy and uphold ethical standards in all stages of the research.

5.2 Reproducibility of Methods

Ensuring the reproducibility of the methods used in this study is fundamental for validating and extending the research findings. All algorithms were implemented in Python 3.9, leveraging the PyTorch framework, and the experimental setups were clearly defined, including the datasets (MNIST, CIFAR-10, Adult, and Abalone) and the specific models evaluated (CGAN, Convolutional ACGAN, CVAE, CTGAN, and TVAE).

To facilitate reproducibility, the source code, including all scripts for data preprocessing, model training, and evaluation metrics, has been made available in a public repository, specified in the abstract of this paper. Detailed instructions for replicating the experiments are provided, ensuring that other

researchers can easily reproduce the results and build upon this work.

By prioritizing ethical considerations and reproducibility, this research aims to contribute responsibly and reliably to the advancement of federated learning and generative model applications.

6 Discussion

6.1 Overview of Findings

The key observations from our experiments reveal nuanced insights into the efficacy and limitations of these models in FL environments. These findings contribute to our understanding of how generative models can be optimized for federated learning, particularly under non-IID conditions.

6.2 Performance of Generative Models in Federated Learning

Our experiments demonstrated that GANs, particularly Conditional GANs (CGANs), exhibit significant variability in performance across different FL setups. In centralized training, CGANs achieved high classification scores, indicating effective generation of realistic data. However, when using federated learning, especially with non-IID data, the performance of CGANs deteriorated. The classification scores remained lower compared to centralized setups, suggesting difficulties in achieving distributional alignment and robust performance in federated environments. This performance drop can mostly be attributed to the inherent challenges posed by non-IID data, which exacerbates the training instability often associated with GANs.

In contrast to GANs, Variational Auto-Encoders (VAEs) displayed more consistent performance across different setups. Conditional VAEs maintained high classification scores and low EMD values, even in federated and non-IID conditions. This robustness can be linked to the probabilistic nature of VAEs, which allows for better handling of data distribution variations across different nodes. The empirical results suggest that VAEs, with their inherent stability and flexibility, are more suitable for federated learning scenarios compared to GANs, particularly when dealing with non-IID data.

6.3 Impact of Data Distribution on Model Performance

The data distribution significantly impacted the performance of all generative models. Centralized setups consistently yielded better results, highlighting the challenges of FL, especially with non-IID data. Federated-IID setups provided a middle ground, where performance was better than non-IID but still not as optimal as centralized training. This gradient in performance underscores the importance of developing techniques to mitigate the effects of Federated Learning.

Addressing Non-IID Data Challenges

Non-IID data distributions pose a major challenge in federated learning by causing bias and reducing the generalizability of the global model. The findings from our experiments suggest that generative models, particularly VAEs, can effectively mitigate some of these challenges. VAEs, with their probabilistic framework, demonstrated robustness and consistency in performance across different federated setups, including non-IID scenarios. This suggests that VAEs can help create synthetic data that better represents the global data distribution, thereby reducing the bias introduced by non-IID data.

Practical Applications and Future Directions

In practical terms, implementing VAEs for data augmentation in federated setups can enhance privacy-preserving data analytics. For instance, in medical applications where data privacy is paramount, these models can generate synthetic patient data that retains the statistical properties of real data without compromising individual privacy. Future research should explore adaptive techniques for integrating generative models into federated learning workflows, optimizing them for various data types and real-world applications. By augmenting local data with high-quality synthetic data, these models can smooth out the discrepancies between different nodes' data distributions, leading to more stable and efficient training processes.

6.4 Limitations and Future Work

While this study provides valuable insights, it also has limitations. The simulation-based FL setup might not fully capture the complexities of real-world FL scenarios. Future research should explore more sophisticated federated learning algorithms and real-world applications to validate and extend these findings. Additionally, training generative models requires a significantly larger amount of data compared to discriminative models. This can be particularly challenging in practical scenarios, such as medical applications where the detection of rare diseases is necessary. Another limitation in this study was the inability to analyse the effects of Federated Learning on diffusion models[11], another promising class of generative models, due to time constraints. Training these models is time-consuming even on specialised hardware, which prevented their inclusion in the analysis.

7 Conclusions

This study explores the impact of federated learning on the performance of various generative models, including GANs and VAEs, under different data distribution scenarios. The key findings indicate that while GANs struggle with stability and performance in federated non-IID environments, VAEs exhibit robust and consistent performance, making them more suitable for federated setups.

In conclusion, this research highlights the promising role of GANs and VAEs in enhancing federated learning through effective data augmentation, paving the way for more robust, generalizable, and privacy-preserving machine learning models.

References

[1] Barry Becker and Ronny Kohavi. Adult. UCI Machine Learning Repository, 1996. DOI: https://doi.org/10.24432/C5XW20.

[2] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, August 2016.

[3] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[4] D.M. Endres and J.E. Schindelin. A new metric for probability distributions. *IEEE Transactions on Information Theory*, 49(7):1858–1860, 2003.

[5] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[6] William Harvey, Saeid Naderiparizi, and Frank Wood. Conditional image generation by conditioning variational auto-encoders, 2022.

[7] Robert Tibshirani Hastie, Trevor J. and Martin J. Wainwright. Statistical learning with sparsity: The lasso and generalizations, 2015.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[9] Mikel Hernadez, Gorka Epelde, Ane Alberdi, Rodrigo Cilla, and Debbie Rankin. Synthetic tabular data evaluation in the health domain covering resemblance, utility, and privacy dimensions. *Methods of Information in Medicine*, 62(S 01):e19–e38, January 2023.

[10] Michael Hilton and Cal. Differential privacy : A historical survey. 2012.

[11] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020.

[12] Maximilian Hoh, Alfred Schöttl, Henry Schaub, and Franz Wenninger. A generative model for anomaly detection in time series data. *Procedia Computer Science*, 200:629–637, 2022. 3rd International Conference on Industry 4.0 and Smart Manufacturing.

[13] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022.

[14] A. Kolmogorov. Sulla determinazione empírica di uma legge di distribuzione. volume 4, pages 83 – 91, 1933.

[15] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research).

[16] H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data, 2023.

[17] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets, 2014.

[18] Sellers Tracy Talbot Simon Cawthorn Andrew Nash, Warwick and Wes Ford. Abalone. UCI Machine Learning Repository, 1995. DOI: https://doi.org/10.24432/C55C7W.

[19] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans, 2017.

[20] Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks, 2015.

[21] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library, 2019.

[22] Karl Pearson. *VII. Note on regression and inheritance in the case of two parents*, volume 58. The Royal Society, December 1895.

[23] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks, 2016.

[24] Yossi Rubner. *Int. J. Comput. Vis.*, 40(2):99–121, 2000.

[25] Abdel Aziz Taha and Allan Hanbury. Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool. *BMC Med. Imaging*, 15(1):29, August 2015.

[26] H. Theil. *Economic Forecasts and Policy*. Number v. 15 in Contributions to economic analysis. North-Holland Publishing Company, 1958.

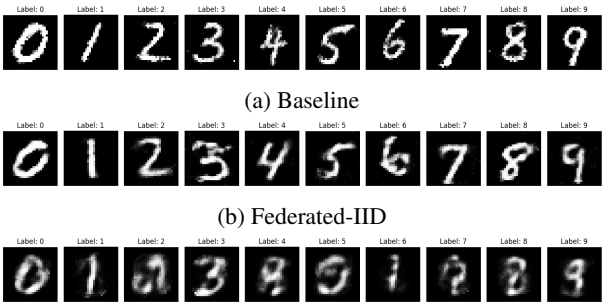
[27] Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. Modeling tabular data using conditional gan, 2019.

[28] Shamim Yazdani, Nripsuta Saxena, Zichong Wang, Yanzhao Wu, and Wenbin Zhang. A comprehensive survey of image and video generative ai: Recent advances, variants, and applications, 01 2024.

[29] Hao Zhang, Qingying Hou, Tingting Wu, Siyao Cheng, and Jie Liu. Data-augmentation-based federated learning. *IEEE Internet of Things Journal*, 10(24):22530–22541, 2023.

[30] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. Federated learning with non-iid data. 2018.

A Appendix

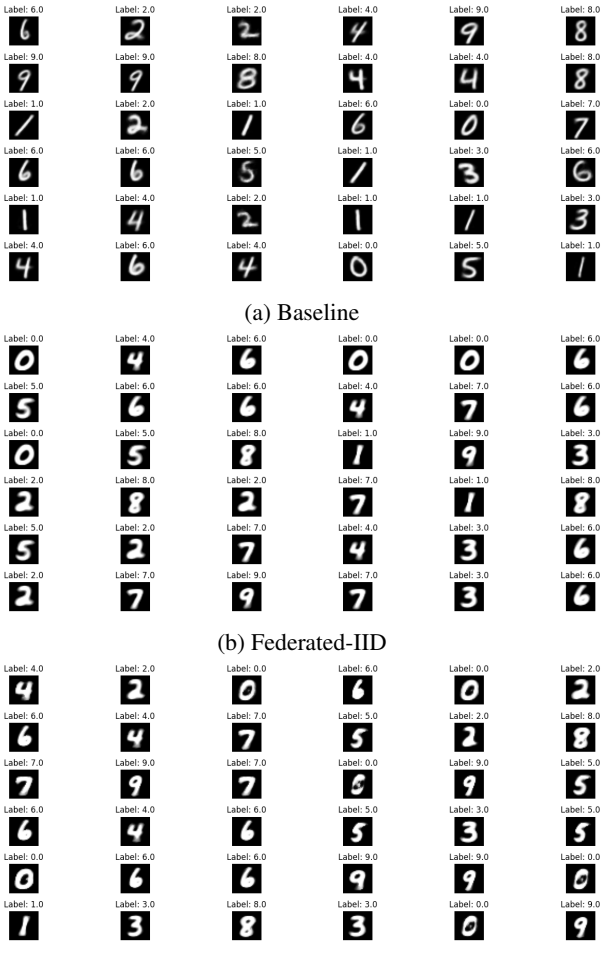


(a) Baseline

(b) Federated-IID

(c) Federted-non-IID

Figure 10: MNIST Samples Generated by the Conditional GAN

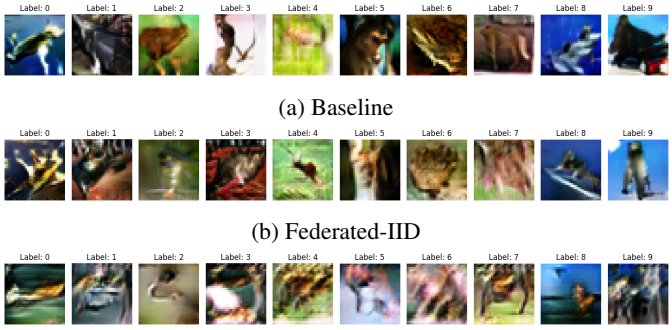


(a) Baseline

(b) Federated-IID

(c) Federted-non-IID

Figure 11: MNIST Samples Generated by the Conditional VAE



(a) Baseline

(b) Federated-IID

(c) Federted-non-IID

Figure 12: CIFAR-10 Samples Generated by the Convolutional AC-GAN