KSME

Springer

Journal of Mechanical Science and Technology 38 (9) 2024

Review Article
DOI 10.1007/s12206-024-0835-0

Keywords:
· Generative model
· Data generation
· Surrogate model
· Deep learning
· Machine learning

Correspondence to:
Dong-Hoon Choi
dhchoi@pidotech.com

Citation:
Kim, D.-K., Ryu, D. H., Lee, Y., Choi, D.-H. (2024). Generative models for tabular data: A review. Journal of Mechanical Science and Technology 38 (9) (2024) 4989~5005.
http://doi.org/10.1007/s12206-024-0835-0

Received February 22nd, 2024
Revised May 20th, 2024
Accepted May 20th, 2024

† Recommended by Editor Hae-Jin Choi

Generative models for tabular data: A review

Dong-Keon Kim, DongHeum Ryu, Yongbin Lee and Dong-Hoon Choi

PIDOTECH Inc., 114, Beobwon-ro, Songpa-gu, Seoul 05854, Korea

Abstract Generative design refers to a methodology that not only simulates the characteristics of a given data or system but also creates artificial data for various purposes. It's a significant research area encompassing diverse issues such as privacy preservation, data distribution analysis, and the development of surrogate models. Initially, research in this field primarily employed stochastic models or basic machine learning methods. However, with the advancement of deep learning technology, numerous studies have emerged, showcasing developed mechanisms using artificial neural network-based methods like variational autoencoders (VAEs) and generative adversarial networks (GANs). These studies extend across different data types, including images and texts, tailored to specific objectives. This paper presents a systematic review of generative design research focused on tabular data. We begin by elucidating the characteristics of tabular data within generative design, followed by a discussion on the goals and challenges in this area. Subsequently, the paper introduces various generative design studies on tabular data, categorized according to their methodological development and unique objectives. Finally, we address the benchmark methods used in generative design for tabular and how their performance is evaluated.

1. Introduction

Recent advancements in artificial intelligence (AI) and machine learning (ML) have emphasized the significance of data across various fields, including computer vision, natural language processing, medicine, finance, mechanical engineering, architectural engineering, transportation, and other fields. Thus, substantial efforts are being devoted to collecting and accumulating data. Notably, extensive datasets such as ImageNet [1] and CelebA [2] in the computer vision field and datasets like WMT [3] and IMDB [4] in natural language processing studies have been openly shared, making significant contributions to the development of deep learning technologies.

Unlike the image and natural language processing fields, however, it can be challenging to collect large volumes of data in specific domains that employ tabular data. This is due to difficulties such as high cost of data generation, sensor malfunctions, human mistakes in recording, or data transmission errors, resulting in partially incomplete datasets. Also, in engineering, obtaining a single data point can often be a time-consuming process. For instance, conducting a computational fluid dynamics (CFD) analysis for complex fluid dynamics systems can take anywhere from a few hours to several weeks to collect a single data point. As a result, securing a significant amount of data is practically infeasible. Furthermore, in fields like healthcare and finance, data collection may be impossible due to privacy concerns, as these datasets often contain sensitive information such as medical records or credit scores. Moreover, data leakage or missing case can lead to erroneous data analysis or poor predictive performance of generated AI models. To address these data scarcity and data quality issues, generative models are widely employed. Generative models use various statistical techniques and artificial intelligence technologies to generate or synthesize information that resembles actual data based on the information available in the original data [5-10].

The aim of this work is to provide an extensive introduction to generative models developed

for the generation of tabular data and a review on these models from various perspectives. Also, we intend to offer guidance to readers who are interested in this field of research. The contributions of this paper are as follows:

1) We introduce foundational knowledge of common terminologies and notations used in the research involving generative models for tabular data. We also present various generative models that can be utilized for generating tabular data and evaluation metrics for assessing the performance of these generative models.

2) We categorize research based on their objectives to allow readers to effectively utilize various generative models for tabular data according to their specific objectives. We provide information regarding the scope of each study including application domains, techniques used, and evaluation methods.

3) We offer guidance to readers who are interested in generative models for tabular data. Additionally, we present current limitations and challenges for further research on generative models for tabular data.

The remainder of this paper is organized as follows. In Sec. 2, we introduce the background knowledge essential for research on generative models for tabular data. In Sec. 3, we categorize and present research on generative models for tabular data based on specific objectives and reflects on the research trends in these studies and summarizes the evaluation results, offering directions for research based on the respective objectives. In Sec. 4, we discuss the limitations of the current research on generative models for tabular data and present the challenges that lie ahead for future researchers. Finally, we provide a summary of our review paper in Sec. 5.

2. Background

In this section, we introduce the prerequisite knowledge necessary for research on generative models for tabular data. The introduced background knowledge encompass:

- Common terminologies and notations used to represent tabular data.
- Various generative models capable of creating tabular data.
- Evaluation metrics employed to assess the performance of generative models.

This fundamental knowledge is intended to aid readers who are new to research on generative models for tabular data in understanding the various studies mentioned in this paper. For those already engaged in related research, it can serve as a review of the essential background information.

2.1 Terminology and notation

Tabular data is structured with rows and columns, typically denoted by m rows and n columns. A table with m rows and n columns is commonly represented as a matrix. Note that columns, often referred to as attributes or variables, hold specific meanings. Attributes are known as design variables in the field of engineering design and as random variables in statistics. Attributes can be categorized into continuous attributes and discrete attributes. A continuous attribute refers to an attribute that can be expressed quantitatively in real numbers, such as weight, frequency, speed, voltage, etc. Most of the research denote continuous attributes as $\mathbf{T} \in \mathbb{R}^{m \times n}$. Note that columns, often referred to as attributes or variables, hold specific meanings. Attributes are known as design variables in the field of engineering design and as random variables in statistics. Attributes can be categorized into continuous attributes and discrete attributes. A continuous attribute refers to an attribute that can be expressed quantitatively in real numbers, such as weight, frequency, speed, voltage, etc. Specifically, continuous attributes are expressed as

$$c_j \in \mathbb{R}^m,\ j \in \{1, 2, \cdots, N_c\}. \tag{1}$$

A discrete attribute is represented as numerical values or strings included in a predefined discrete set specific to each discrete attribute. Discrete attributes are denoted as

$$d_j \in \mathbb{Z}^m,\ j \in \{1, 2, \cdots, N_d\}. \tag{2}$$

Discrete attributes can be further classified into binary attributes, nominal attributes and ordinal attributes. Binary attributes refer to attributes that are expressed as 0 or 1, representing the presence or absence of a certain condition or the existence of a particular object. Nominal attributes represent attributes that have no inherent order or physical significance, such as colors or blood types. On the other hand, ordinal attributes indicate attributes that contain an inherent order, such as ratings. To train generative models, the nominal and ordinal attributes are often converted into discrete attributes using embedding techniques.

Tabular data typically exists in a form where various types of attributes are mixed. In tabular data, rows represent individual data instances, and each row is described by the values of multiple attributes. The ith individual instance is generally represented as shown in Eq. (3).

$$R_i = \{c_{1,i}, c_{2,i}, ..., c_{N_c,i}, d_{1,i}, d_{2,i}, ..., d_{N_d,i}\} \in \mathbb{R}^{N_c+N_d},\quad \text{where } i \in \{1, 2, \cdots, m\}. \tag{3}$$

Eventually, tabular data with N_c continuous attributes and N_d discrete attributes is typically represented as expressed in Eq. (4).

$$T = \{r_1, r_2, ..., r_m\}^T. \tag{4}$$

2.2 Basic model

Generative models for tabular data have evolved with the advancement of statistical techniques and artificial intelligence. As illustrated in Fig. 1, prior to the development of models de-

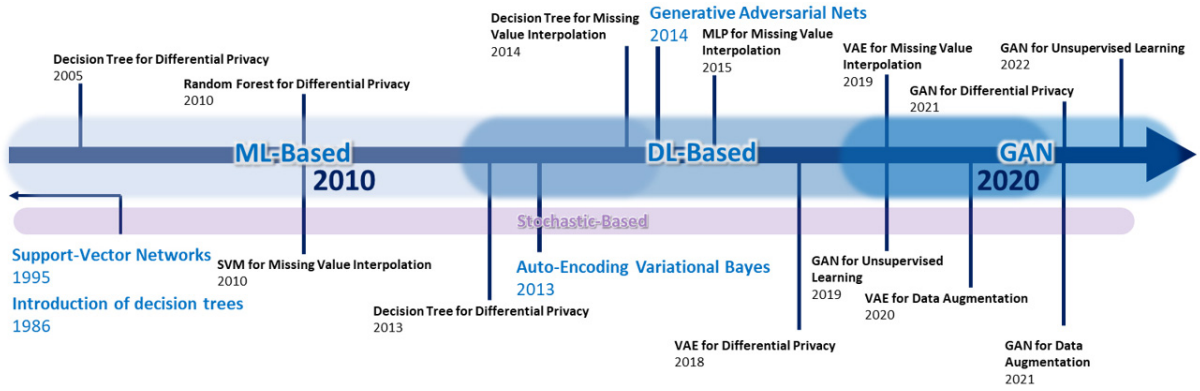


Fig. 1. Generative models for tabular data.

signed exclusively for data generation such as variational autoencoder (VAE) and generative adversarial network (GAN), inference models like decision tree (DT), random forest (RF), support vector machine (SVM), and multi-layer perceptron (MLP) were commonly utilized. Even though they are not mainstream models, stochastic models such as Gaussian process and Bayesian network has also been used for generating data.

Since 2010, with the rapid progress in computing power and artificial intelligence technology, a variety of generative models have been proposed. Kingma and Welling introduced VAE in 2013 [5], and L. Goodfellow proposed GAN in 2014 [8]. These two models can generate data that is realistic and closely resembles the distribution of given data. As a result, they have been applied across various fields and have gained recognition for their capabilities. Recently, they have also been widely utilized for generating tabular data.

In this paper, we aim to introduce various foundational models used for generating tabular data, classified as machine learning (ML)-based models, deep learning (DL)-based models, GAN models, and stochastic models.

2.2.1 Machine learning-based models

In 2000s, research on generative models relied primarily on machine learning models such as decision tree (DT), random forest (RF), and support vector machine (SVM). These methods predominantly comprised inference models [9]. ML models exhibited an advantage over conventional stochastic methods by showing satisfactory performance even with numerous attributes, which contributed to their widespread utilization for generating tabular data. These inference models can generate new data by inferring values for arbitrary instances.

2.2.1.1 Decision tree

Decision tree, also known as classification and regression tree (CART), is a model that splits data into different conditions to make inferences by minimizing entropy, as illustrated in Fig. 2 [8]. DT is known for its intuitiveness in presenting branching criteria, making it a well-recognized explainable model. However, DT is limited by the fact that it derives inferences through

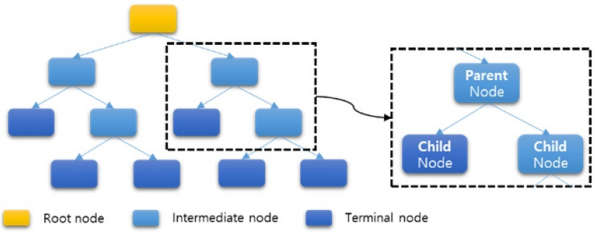


Fig. 2. Illustration of tabular data notation.

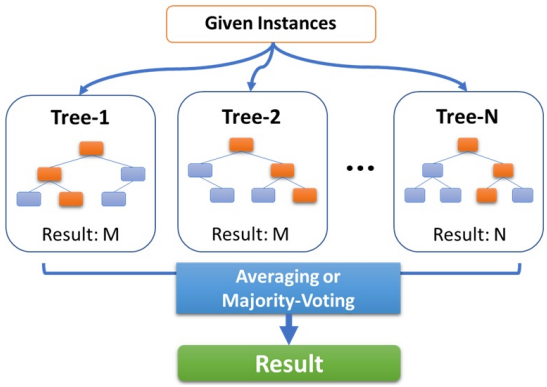


Fig. 3. Structure of random forest with N tree models.

a single learner. When attempting to minimize the entropy-based loss function over training data to an extreme degree, there is a risk of overfitting, which can lead to reduced generalization performance. Although overfitting issues can be addressed through pruning, achieving a significant improvement in inference performance is fundamentally challenging when applied to a single learner.

2.2.1.2 Random forest

The problems with DT, as described earlier, can be addressed by assembling various learners. A prominent ensemble model is random forest (RF). RF was developed to mitigate the issue of trade-offs between overfitting and generalization performance encountered by DT. It involves creating multiple weak learners and aggregating their results, as shown in Fig. 3.

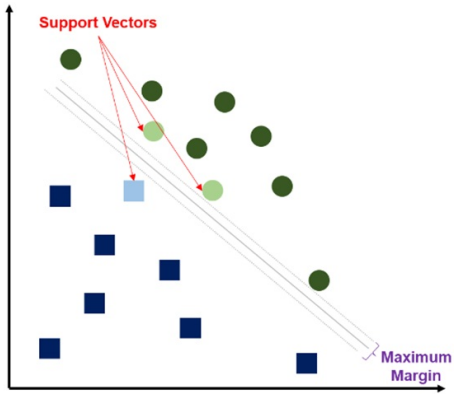


Fig. 4. An illustrated example of support vector machine.

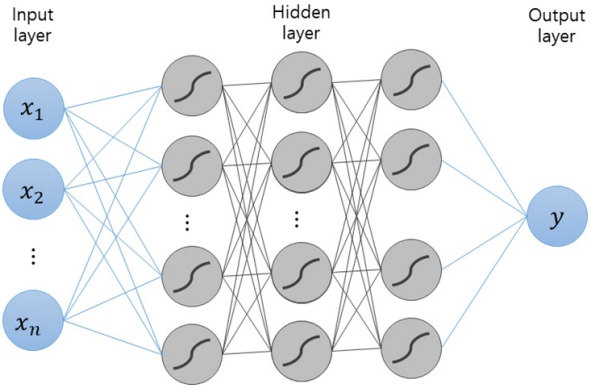


Fig. 5. Basic MLP structure.

To introduce diversity among the weak learners, random subsets of attributes from the given data are utilized during the creation of each weak learner. These characteristics contribute to the robust inference performance of RF compared to that of DT, exhibiting better generalization capabilities. Nevertheless, ensemble models like RF have inherent limitations. Specifically, since they average the results obtained from multiple learners, they cannot achieve the highest level of regression performance when used for predictive purposes.

2.2.1.3 Support vector machine

Support vector machine (SVM) is a model that creates a hyperplane, which can classify data in high dimensions through a kernel function. Its goal is to learn a hyperplane that maximizes the margin, which is the sum of distances between support vectors and the hyperplane [7], as shown in Fig. 4. SVM offers advantages of being less susceptible to the influence of erroneous data and a reduced risk of overfitting by permitting some margin of error. However, SVM comes with various hyperparameters for constructing the model, including the soft margin parameter, kernel type, and the gamma parameter that determines the shape of the kernel. Finding the optimal values of these hyperparameters can be time-consuming as they can vary depending on data characteristics. Furthermore, SVM is known for its slow training speed, and unlike DT-based models, it constructs a complex black-box model that is challenging to interpret.

2.2.2 Deep learning-based models

During the 2010s, with the advancement of computer hardware technology, various deep learning models saw a tremendous surge. Following this trend, deep learning models were extensively adjusted for generating tabular data, resulting in models that exhibited improved generative performance compared to ML-based models [1]. Deep learning models for tabular data generation primarily utilize artificial neural networks. Unlike ML-based techniques, deep learning methods provide flexibility in modifying the neural network structure, including the number of hidden layers, node counts, and activation functions, allowing the creation of models tailored to data features. Due to these advantages, DL-based approaches became capable of better analyzing complex relationships between attributes, allowing for the generation of realistic tabular data that maintains the distribution of existing data. As a result, research in various fields actively progressed in the use of deep learning models as generative models for creating tabular data.

2.2.2.1 Multi-layer perceptron

Multi-layer perceptron (MLP), as shown in Fig. 5, is a model with multiple perceptrons connected through multiple hidden layers. The advantage of the MLP is that, as the number of the hidden layers increases, it can better interpret complex data relationship and represent the nonlinearity of the system. Like the ML-based methods mentioned earlier, MLP is an inference model and comes with a considerable number of hyperparameters including network architecture, activation function, optimizer, learning rate, and more. The optimal hyperparameters differ based on the characteristics of data. Additionally, the performance of MLP is sensitive to hyperparameters, introducing a level of complexity that can be challenging for individuals without expertise in the field. While there are techniques for optimizing hyperparameters of MLP, it remains a field with ongoing challenges and room for improvement [11].

2.2.2.2 Variational autoencoder

Variational autoencoder (VAE) shares a structure like MLP and autoencoder (AE) [5], but it is fundamentally different from the MLP as it is designed solely for data generation, as seen in Fig. 6. Like AE, VAE has an encoder and decoder structure. However, VAE compresses the distribution information of the given training data into a lower-dimensional parameterization during the training process. This allows VAE to generate new data that closely resembles the actual data by considering the distribution of the training data. It is important to note that AE is fundamentally a manifold learning method, while VAE is specifically designed for data generation. Readers are encouraged to distinguish between AE and VAE and use them for their respective purposes. Furthermore, VAE is widely utilized in generating tabular data due to the advantage of handling smaller-dimensional data, which is favorable compared to im-

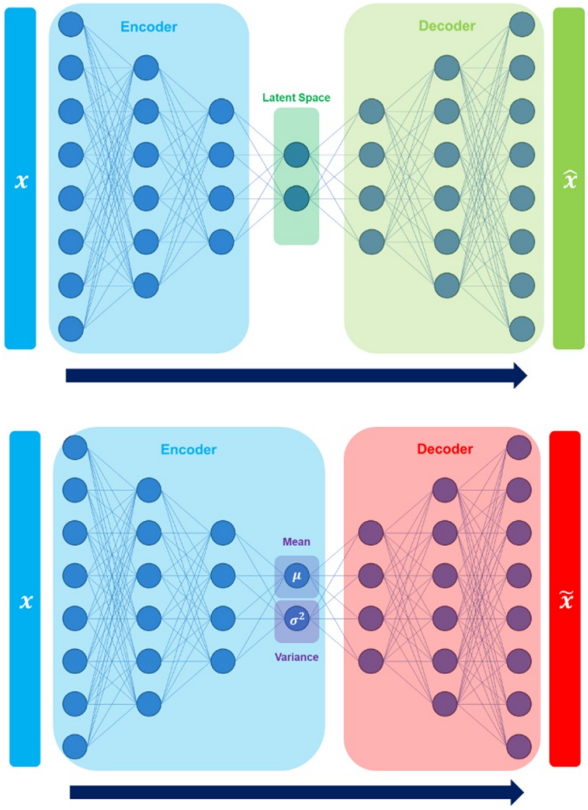


Fig. 6. Autoencoder (top) and variational autoencoder (bottom).

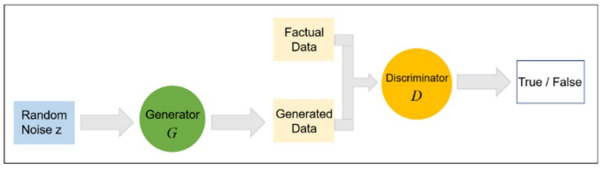


Fig. 7. GAN's generator (G) and discriminator (D).

age-text data [12-15].

2.2.3 Generative adversarial network (GAN)

Generative adversarial network (GAN) is a more advanced generative model than the VAE model, as depicted in Fig. 7. It consists of two neural network components: the generator and the discriminator. The generator is a neural network structure responsible for generating data, while the discriminator is the other neural network that assesses the authenticity of the input data. The discriminator is trained to determine the authenticity of data, and the generator is trained to deceive the discriminator, making the generated data appear as if it were in the real world. This structure allows GANs to generate highly realistic data. While GANs have been primarily used in the computer vision field for synthesizing and generating image data, recent research has proposed using GANs for generating tabular data as well [14-16].

2.2.4 Stochastic models

Before the active use of machine learning and deep learning for generating tabular data, the field of statistics predominantly employed probability-based inference models. Notable examples include Gaussian process and Bayesian network. These inference models are favored for their mathematical representability, allowing for intuitive interpretation and analysis.

2.2.4.1 Gaussian process

Gaussian process (GP) is a type of stochastic process, where a linear combination of finitely observed data follows a multivariate normal distribution. GP represents deterministic values at observed locations and probabilistic values conforming to a multivariate normal distribution at unobserved locations. Additionally, GP is advantageous for analyzing systems with strong non-linearity as it considers the correlations among data points within the domain where data exist. However, to train a GP model, it is essential to solve an inverse matrix problem that scales with the amount of given data. Therefore, GP models can only be utilized effectively in cases where the available data is quite limited.

2.2.4.2 Bayesian network

Bayesian network is a probabilistic model that represents the relationship between data as conditionally independent through a directed acyclic graph and performs well when there exists a clear causal relationship between data points. One of its advantages is the intuitive interpretability of the model since it uses probabilities to indicate the likelihood of specific values occurring.

2.3 Evaluation metrics

There are various metrics for evaluating generative models for tabular data, and these evaluation metrics are chosen based on the specific objective of each study. In this section, various metrics used for evaluating generative models are to be introduced.

2.3.1 Correlation coefficient

Correlation coefficient, also known as the Pearson correlation coefficient, is a metric that quantifies the linear correlation between two variables. It takes values between -1 and 1, where a larger absolute value indicates a stronger linear relationship between the two variables. This metric can be employed when a quantitative evaluation of the degree of similarity between the generated data and the real data is required. The correlation coefficient between the ith row $\mathbf{r}_i$ in the table $\mathbf{T}$ and the row $\hat{\mathbf{r}}_i$ created by the generative model with $\mathbf{r}_i$ set as the target answer is computed by the Eq. (5).

$$\mathrm{Corr}_i = \frac{n\sum_{j=1}^{n} x_{j,i}\hat{x}_{j,i} - \sum_{j=1}^{n} x_{j,i}\sum_{j=1}^{n} \hat{x}_{j,i}}{\sqrt{n\sum_{j=1}^{n} x_{j,i}^2 - \left(\sum_{j=1}^{n} x_{j,i}\right)^2}}. \quad (5)$$

Here, $x_{j,i}$ is the jth variable of $\mathbf{r}_i$, and $\hat{x}_{j,i}$ refers the jth variable of $\hat{\mathbf{r}}_i$. Notation n indicates the number of the variables in the row. When multiple rows need to be generated, the final correlation coefficient is calculated by averaging the values of the individual correlation coefficient computed for each row.

2.3.2 ε-loss

ε-loss is a metric used in the field of differential privacy to quantify the extent to which data privacy is preserved. A smaller ε-loss value indicates that data confidentiality is preserved more effectively. This metric is used to verify whether the generated or substituted data can infer specific values from the actual data. The method to calculate ε for the original table $\mathbf{T}$ and a table $\mathbf{T}'$ with only 5 % of its rows replaced is as follows:

$$\frac{\Pr(\mathbf{T} \in \text{Real})}{\Pr(\mathbf{T}' \in \text{Real})} \leq e^{\varepsilon}. \tag{6}$$

2.3.3 F1-score

F1-score is a metric used to measure the predictive performance in classification tasks, indicating how well the predicted discrete values match with the ground truth. The F1-score serves as a performance metric to evaluate the effectiveness of the classification task. The F1-score is expressed in terms of precision and recall, as shown in Eq. (7).

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{7}$$

Precision and recall are represented by Eqs. (8) and (9), respectively.

$$\text{Precision} = \frac{(\#\text{ of TP})}{(\#\text{ of TP}) + (\#\text{ of FT})} \tag{8}$$

$$\text{Recall} = \frac{(\#\text{ of TP})}{(\#\text{ of TP}) + (\#\text{ of FN})}. \tag{9}$$

Note that TP, FP, and FN indicate true positive, false positive, false negative, respectively.

2.3.4 Rooted mean squared error

Rooted mean squared error (RMSE) is indeed a common metric for regression tasks that measures the average magnitude of the differences between predicted and actual values. Lower RMSE values indicate better predictive performance of the model. It's used in regression tasks to evaluate how well a model's predictions align with the actual continuous values. The RMSE between the ith row $\mathbf{r}_i$ in the table $\mathbf{T}$ and the row $\hat{\mathbf{r}}_i$ created by the generative model with $\mathbf{r}_i$ set as the target answer is computed by Eq. (10).

$$\text{RMSE}_i = \sqrt{\sum_{j=1}^{n} \frac{(\hat{x}_{j,i} - x_{j,i})}{n}}. \tag{10}$$

Note that $x_{j,i}$ is the jth variable of $\mathbf{r}_i$, and $\hat{x}_{j,i}$ denotes the jth variable of $\hat{\mathbf{r}}_i$. Notation n indicates the number of the variables of the row. When multiple rows need to be generated, the final RMSE value is gathered by averaging the individual RMSE values calculated for each row.

2.3.5 Kolmogorov-Smirnov test

Kolmogorov-Smirnov test (KS-test) is a metric used to determine whether two sample sets are drawn from the same population. Specifically, it examines the similarity of the cumulative probability distributions of the two sets through a test statistic. This is useful for assessing the similarity of distributions between generated data and real data. In this context, for continuous attributes, it leverages the cumulative distribution function (CDF), and for discrete attributes, it employs the chi-squared test for verification. The test statistic for the KS-test is expressed as shown in Eq. (11) with the given distribution of the original data F, the distribution of the generated data F', and the total number of rows m in the table.

$$D_{m,m} = \sup_x |F_m(x) - F'_m(x)|. \tag{11}$$

2.3.6 Kullback–Leibler divergence

Kullback–Leibler divergence (KL-divergence) is a metric that quantifies the difference between two probability distributions. A smaller KL-divergence value indicates that the two distributions are more similar. Unlike the KS-test, which expresses the metric using a test statistic, KL-divergence quantifies the "information difference" from one distribution to another. This metric is also used to assess the similarity between the distribution of generated data and real data. For an actual data distribution Q and the distribution of data generated by the generative model P, KL-divergence is defined as shown in Eq. (12).

$$KL(P \| Q) = \sum_x P(x) \log \frac{P(x)}{Q(x)}. \tag{12}$$

3. Research status and guides

In Sec. 2, various models for generating tabular data were introduced. These models are being utilized for research on table data generation in various fields, each serving its specific purpose. Based on our investigation, generative models for tabular data primarily serve four distinct purposes: differential privacy, missing value interpolation, data distribution modeling, and data augmentation.

In this section, we introduce various studies on generative models for tabular data, each of which focuses on specific objectives. We then described evaluation methods presented in these studies. We included only those studies that have been referenced at least once by other studies, to avoid unacknowledged research. Furthermore, we offer guidance on the future research directions for those interested in developing generative models for tabular data.

3.1 Differential privacy

Generative models for differential privacy are primarily utilized in fields like finance and healthcare, where sensitive information is involved. Data containing such privacy concerns are often inaccessible directly. Thus, generative models are constructed based on the original data to generate realistic data without sensitive information. Early research in tabular data generation mainly focused on designing generative models for the purpose of achieving differential privacy. As a result, a wide range of generative models, from ML-based models to GANs, has been employed in this context.

The first generative models designed for the purpose of achieving differential privacy were primarily applied in the medical field. In 2005, Reiter [17] proposed a framework for generating data using DT models. In the medical domain, certain attributes of tabular data may contain sensitive information, making arbitrary small changes to only the privacy-sensitive attributes instead of the entire dataset. In this research, DT model was trained with the given dataset to generate non-sensitive attributes using privacy-sensitive ones as the input. However, as mentioned in Sec. 2, DT has limitations due to the use of a single weak learner, potentially causing negative impacts on generalization and robustness. In 2010, Caiola and Reiter [18] addressed the generalization and robustness issues by utilizing RF in a generative model. This study employed RF to generate data in a chained manner, using a subset of attributes to generate the target attribute.

In 2018, Sun et al. [19] proposed a generative model that combined MLP with the vine copula technique [20]. The neural network structure of the proposed generative model resembled the decoder part of VAE, and the model's training incorporated concepts from reinforcement learning. In 2019, Li et al. [21] used VAE as a generative model to create artificial medical data that closely resembled real data.

In 2017, Choi et al. [22] introduced a GAN model for generating artificial electronic health records (EHRs), marking the first utilization of GAN for differential privacy. The proposed GAN model had a structure similar to AE, training the original data with an encoder-decoder structure. The generated data underwent a classification process with a discriminator, focusing on the similarity between generated and original data rather than concerns about data information loss. In 2019, Baowaly et al. [18] proposed a method to preserve the differential privacy of medical records. They proposed a GAN model capable of handling data with irregular distributions and class imbalances. This paper first trained the original data in an autoencoder structure and then employed the decoder part of the trained model as the generator in the GAN. In 2019, Chen et al. [20] aimed to enhance generative performance by introducing a Wasserstein GAN (WGAN)-based generative model. The authors replaced the loss function of Choi et al.'s model [22] with Wasserstein [23] and introduced the boundary condition concept [24]. This enabled the creation of a stable learning structure for both continuous and discrete attributes, enhancing the model's stability. In 2020, Rashidian et al. [25] performed data preprocessing to generate data like real data. They preprocessed given EHRs using smoothing techniques and fed the preprocessed data into a WGAN-based model to create new EHR data.

Goncalves et al. [26] in 2020 and Kaur et al. [25] in 2021 conducted research that employed stochastic methods for medical data's differential privacy. Goncalves et al. [26] proposed a Gaussian process model capable of handling multimodal distributions, including both continuous and discrete attributes. They especially addressed categorical attributes through the product of multinomials. Kaur et al. [25] designed a generative model using a graph-structured Bayesian network. The trained Bayesian model replaced the desired sensitive information with arbitrary data to protect the information.

Generative models designed for the purpose of achieving differential privacy have been actively used not only in the medical field but also in various other domains. In the field of demographics, sensitive privacy data such as personal information are present, making generative models crucial for preserving differential privacy.

In 2010, Drechsler [27] proposed an SVM-based generative model for achieving differential privacy in the demographics field. The author customized the loss function to adapt the generative model to the given data environment.

In 2018, Tai et al. [28] introduced a VAE model considering data confidentiality and dimension reduction in the demographic domain. The paper focused on learning the representation of the original data to reduce dimensions and proposed a model that generates replacement values for secure data using the reduced dimensions.

In 2018, Park et al. [29] conducted research to improve upon Choi et al.'s technique [22], aiming to create tables that protect privacy without compromising data integrity. They constructed a framework to ensure that machine learning models trained on data generated by GAN produced similar performance to ML models trained on the original data. The GAN generator synthesized data using convolution transpose structures [30]. In 2020, Walia et al. [31] also worked on enhancing the training stability of generative models and, similar to Chen et al. [20] in 2019, utilized Wasserstein loss. However, unlike Chen et al.'s research, they added a gradient penalty term to improve the model's training stability, which allowed for unbiased data generation across modes. In 2020, Zhao et al. [32] performed preprocessing, considering the characteristics and distributions of tabular data, used the Gaussian mixture model (GMM) [33] to interpret multimodal data distributions, and proposed a model that embedded the distribution information obtained from GMM. Notably, they included the parameters used in the embedding part of the GAN training process for optimization.

In 2014, Zhang et al. [34] proposed a Bayesian network for generating similar distributions to the given data. They designed Bayesian network represented relationships for each attribute in a non-cyclic graph, where each connection was expressed as conditional probability. They introduced a

model that added noise to the Bayesian network learned from the given data to create artificial mimic distributions, stating that this method can be applied to datasets with many attributes.

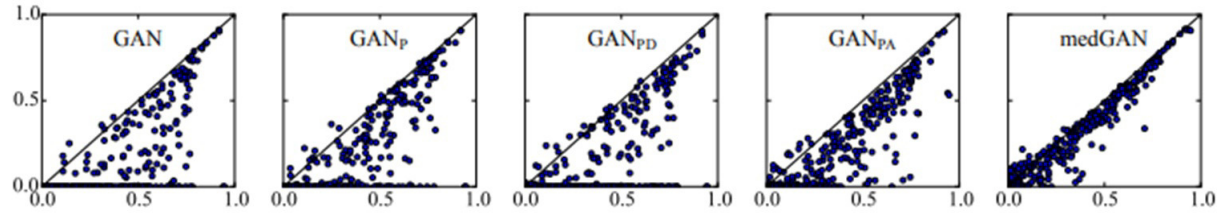
Additionally, generative models for differential privacy have found applications in the business, engineering, and transportation fields. In 2013, Lee et al. [35] proposed a DT-based generative model for ensuring differential privacy in the business domain. In 2021, Tai et al. [7] introduced a VAE framework for preventing data leaks in the business field. They proposed methods for training the model differently, depending on various scenarios related to data loss. In 2018, Mottini et al. [36] presented a GAN model aimed at protecting passenger information containing personal attributes in the transportation domain. They analyzed data including missing values and outliers, and proposed a model based on Cramer GAN [37] for generating such data. In 2018, Yoon et al. [15] applied a GAN model for achieving differential privacy in the field of engineering. While previous data generation research mainly focused on generating realistic data, Yoon et al. [15] focused on secondary tasks such as classification or regression. The paper introduced a teacher-student structured discriminator constructed sequentially to ensure that the generator could create realistic data, which led to excellent predictive performance of the classification and regression models. However, a limitation of this model was that it considered only ideal data distributions such as uniform and normal distributions.

In the context of differential privacy, the primary performance metrics of interest revolve around assessing the similarity between the generated data and the original data distribution, as well as preserving data confidentiality. The quality of generated data is considered excellent if it closely resembles the original data distribution, and the assessment of such similarity often hinges on evaluating the similarity of distributions. Furthermore, for the purpose of ensuring differential privacy, the generated data should make it challenging to infer the original data, thus ensuring strong confidentiality.

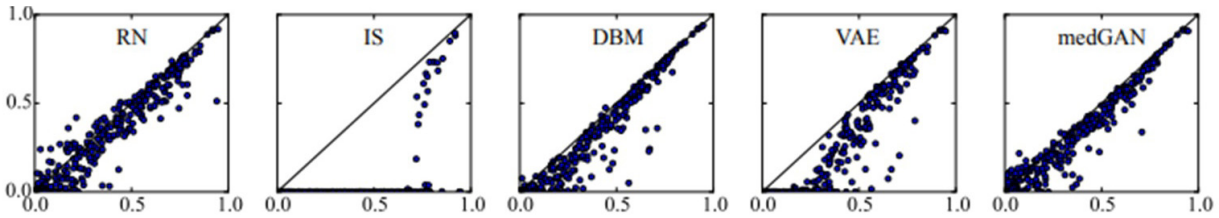
As shown in Table 1, the main performance metrics used to

Table 1. Evaluation results from differential privacy researches.

Type	Data set	Corr	ε - loss	F1-score	Figure
Public	Adult	Ref. [31], DL: 0.8527	Ref. [38], GAN: 0.3200 Ref. [39], Stochastic: 0.2200	Ref. [37], GAN: 0.8310	Ref. [28], DL: Dist [Fig. 10.]-(a) Ref. [33], DL: Dist [Fig. 10.]-(b)
	Census-income	Ref. [40], GAN: 0.6780 Ref. [41], GAN: 0.6930	-	-	-
	ISOLET	Ref. [9], GAN: 0.6399	-	-	-
	Heart disease	-	-	-	-
	Credit	Ref. [41], GAN: 0.9190	-	-	-
	Convertype	-	-	Ref. [37], GAN: 0.4970	-
	Health	-	Ref. [38], GAN: 0.0300	-	-
	Airline	-	Ref. [38], GAN: 0.0900	-	-
	13 datasets of UCI (avg.)	-	Ref. [42], GAN: 0.2090	-	-
	Sugar farms dataset	Ref. [3], ML: 0.8820	-	-	-
	Wisconsin breast cancer	Ref. [43], DL: 0.9100	-	-	-
	PAMF	-	-	Ref. [37], GAN	-
	LACity	-	Ref. [38], GAN: 0.1900	-	-
	MIMIC-III	Ref. [37], GAN: 0.9794	-	-	-
	Cerner health facts database	-	Ref. [44], GAN: 0.2400	-	-
	SEER's research dataset	Ref. [45], stochastic: 0.7600	-	-	-
	Bank marketing data	Ref. [31], DL: 0.9684	-	-	-
	Extended MIMIC-III	Ref. [10], GAN: 0.9724	-	-	-
	Air carrier statistics	Ref. [40], GAN: 0.9830	-	-	-
	NLTCS	-	Ref. [39], Stochastic: 0.2700	-	-
	Cardiovascular	Ref. [41], GAN: 0.7410	-	-	-
Custom	U.S. current population survey	Ref. [2], ML: 0.9380 Ref. [4], ML: 0.7720	-	-	-
	IAB establishment panel	-	Ref. [46], ML: 0.2240	-	-
	Airline dataset	-	-	-	Ref. [47], GAN: Dist [Fig, 10.]-(c)

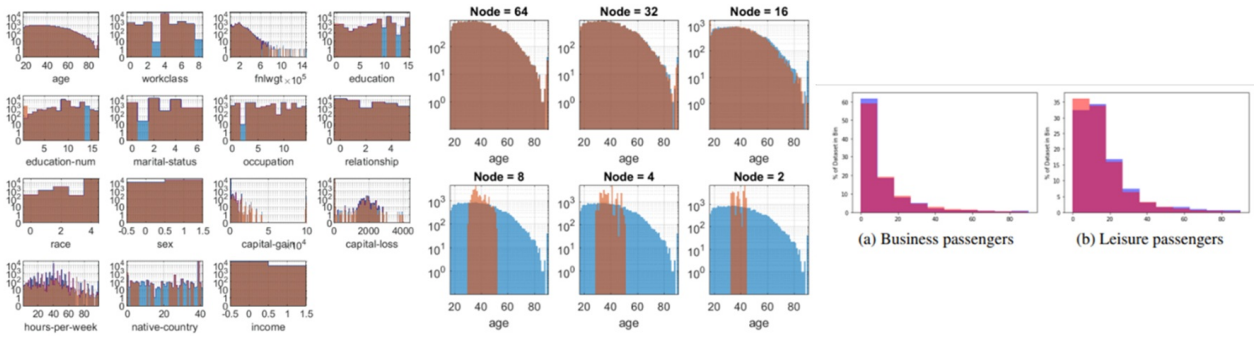


(a) Dimension-wise prediction performance of various versions of medGAN



(b) Dimension-wise prediction performance of baseline models and medGAN

Fig. 8. Distribution of generated data from the method in Ref. [22].



(a) (b) (c)

Fig. 9. Distribution of generated data from the method in Ref. [22].

evaluate generative models in the context of differential privacy are typically correlation coefficient and ε-loss. The correlation coefficient is a versatile metric widely used for various purposes, while ε-loss is primarily employed in studies that focus on data confidentiality. Additionally, some studies, such as Choi et al. [22], build classification models using the generated data to assess inference performance. They evaluated performance using the F1-score of a logistic regression model trained on generated data. Furthermore, as depicted in Fig. 9, visualizing the distribution of generated data and comparing it to the original data distribution can provide a visual means to assess the similarity between the two.

Many of the early studies, with a focus on differential privacy, primarily employed GANs, as can be observed in Table 1. This could be attributed to GANs being the most advanced type of generative model available, and their superior ability to generate realistic data compared to other models, as indirectly evidenced by the performance metrics in Table 1. Therefore, for future researchers aiming to develop generative models for tabular data with a focus on differential privacy, it is recommended to consider utilizing generative models based on GANs tailored to the specific domain.

3.2 Missing value interpolation

Generative models serve another crucial purpose: interpolating missing data. In the early stages of research on generating tabular data, studies were conducted to develop generative models with the purpose of interpolating missing data, alongside the objective of differential privacy. In reality, there are often various reasons for missing data during data collection. If a substantial portion of the data is missing, it can negatively impact data analysis and the design of inference models. Therefore, interpolating missing data is a vital issue. However, real-world data typically exhibit correlations among attributes, making it challenging to replace missing values using simple linear interpolation methods. Hence, researchers have proposed methods to interpolate missing data using generative models that consider all the characteristics of tabular data [38, 47]. The issue with missing data in tabular datasets is pervasive in various fields, thus the challenge of interpolating missing data needs to be addressed, where generative models are increasingly being applied as one of the potential solutions.

In the demographic field, missing data can occur due to no responses from survey participants. Burgette and Reiter [48] in

2010 and Doove et al. [49] in 2014 employed decision trees (DTs) to interpolate such missing data that could arise in the demographic domain. Burgette and Reiter [48] conducted an analysis of correlations among all attributes and proposed a model to generate attribute values with missing ones. On the other hand, Doove et al. [49] introduced a DT model capable of multiple interpolations and incorporated the multiple imputation by chained equations (MICE) algorithm, allowing for the preservation of the distribution of the training data while learning the DT model.

In the field of engineering, missing data can occur due to equipment malfunctions or miscalibrations. In 2015, Valdiviezo et al. [40] proposed a data-driven decision-making framework based on DTs for handling missing data that could arise from difficulties in accessing information in the field of computer science. In 2018, McCoy et al. [50] introduced research aimed at addressing the issue of missing data resulting from equipment malfunctions in a simulated milling circuit. They utilized a VAE model to address missing data problems. To train the VAE model, they randomly omitted parts of the given training data and designed a model to train the proposed VAE model to restore the missing portions from the original data. In 2015, Yan et al. [51] proposed a generative model for interpolating missing data in the context of IoT data transmission. They interpreted the distribution of the existing data using a Gaussian mixture model and used this information to interpolate the missing values.

In the medical field, missing data problems can arise due to issues like medical equipment malfunctioning, lack of monitoring, or incorrect measurements. In 2017, Xia et al. [41] proposed an algorithm based on random forest (RF) to address missing data issues in the medical domain. Specifically, they introduced the adjusted weight voting random forest (AWVRF) algorithm, which could interpret attributes without interpolating missing values. In 2015, Purwar et al. [52] proposed a tabular data generation framework that utilized clustering techniques to address missing data problems. They used 11 different missing data interpolation methods, including traditional statistical approaches and weighted K-nearest neighbor (WKNN), a form of unsupervised learning. They created 11 datasets, each with a different missing data interpolation method, and selected the dataset with the best cluster validation. In 2020, Akrami et al. [38] introduced a data generation method using a variation of VAE known as β-VAE. This approach was customized to fit the needs of tabular data generation with tailored loss functions.

Besides, in the transportation field, research utilizing generative models to address missing data interpolation problems has been investigated. In 2019, Bouquet et al. [47] proposed an asymmetric VAE, which was distinct from the typical symmetric encoder-decoder structure of VAE models. The proposed model introduced a method for both interpolation of missing data and performing a secondary task, such as regression, using interpolation data and a multi-layer perceptron (MLP). In the same year, Zhao et al. [52] proposed a stochastic-based generative model with the aim of missing data interpolation and utilized Gaussian copula to interpolate missing values, not only for continuous attributes but also for binary attributes. Notably, for discrete attributes, the model employed a cutoff function (step function) to interpolate missing values.

In the context of addressing missing data interpolation problems, key evaluation metrics include the correlation coefficient, F1-Score and RMSE. Like the case of differential privacy, when it comes to ensuring that generated data closely resembles real-world data, the key evaluation metric used is the correlation coefficient. Obtaining high similarity between the distributions of the generated data and the original data is crucial in maintaining the realism of the generated data. Furthermore, since missing data interpolation problems often involve the application of inference models, performance is also assessed using F1-score and RMSE as evaluation metrics. F1-score is applied when dealing with classification tasks, while RMSE is used for regression tasks. Additionally, evaluation of generative models is sometimes carried out using methods like those shown in Figs. 10 and 11. Fig. 10 provides a performance comparison between VAE proposed by Akrami et al. [38] and the standard VAE. These models were evaluated by applying datasets generated through missing data interpolation to secondary tasks, such as inference models, and observing the loss values during training. In this case, the x-axis represents the rate of interpolated missing data, and the y-axis represents the loss value. x = 0 indicates datasets where no missing data was interpolated, while x = 1 indicates datasets where all missing data was interpolated. Fig. 10 illustrates that data generated from VAE proposed by Akrami et al. showed more stability during model training compared to the standard VAE. Next, Fig.

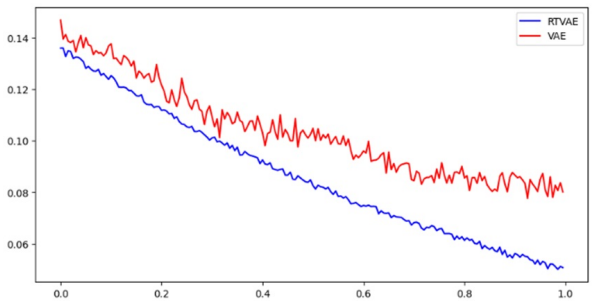


Fig. 10. Loss reduction graph of the proposed generative model in Ref. [38] and the ordinary VAE model.

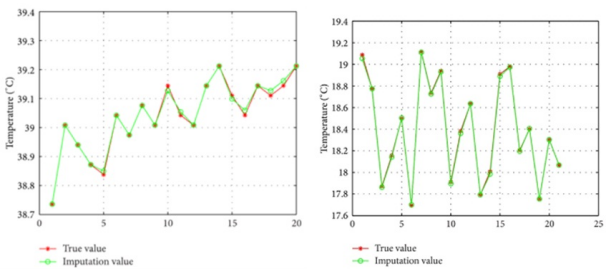


Fig. 11. Data imputation results from Ref. [52]. This graph shows the interpolated results for two data variables.

Table 2. Evaluation results from differential privacy researches.

Type	Data set	Corr	F1-score	RMSE	Figure
Public	Disorders	Ref. [53], ML: 0.6839	-	-	
	Pima indians	-	Ref. [26], ML: 0.9982	-	
	Heart disease	Ref. [53], ML: 0.8776	-	Ref. [19], ML: 0.0800	
	Diabetes	Ref. [53], ML: 0.7898	-	-	
	Cancer	Ref. [53], ML: 0.9912	Ref. [26], ML: 0.9939	-	
	Hepatitis		Ref. [26], ML: 0.9908	-	
	Mi data	-	-	Ref. [17], ML: 0.1980 Ref. [18], ML: 0.1100	
	KDDCup 99	-	-	-	Ref. [11], DL: loss [Fig. 11]
	Caltrans PeMS	-	-	Ref. [21], DL: 0.0918	
Custom	Haberman's survival dataset	-	-	Ref. [19], ML: 0.0500	
	Simulated milling circuit dataset	-	-	Ref. [32], DL: 0.8440	
	Intel berkeley research lab		-	-	Ref. [54], stochastic: inference [Fig. 12]
	General social survey	-	-	Ref. [55], stochastic: 0.8770	

11 shows the results from Zhao et al. [52], where the results of interpolating temperature data recorded at various sensors over time with the actual data were compared and found little difference.

In the missing data interpolation task, which involves filling in missing values for specific attributes, it may be a relatively easier task compared to other objectives. Therefore, as seen in Table 2, simpler inference models like DT and MLP are predominantly used rather than complex models like GAN. Moreover, the per formance of generative models can significantly vary depending on the given dataset. Hence, it is advisable for readers researching generative models for the purpose of missing data interpolation to explore various inference models and choose the one that performs best for their specific dataset and requirements.

3.3 Data distribution modeling

Research using generative models to simulate data distributions is relatively scarce due to the difficulty of the task compared to other research areas. In the early 2000s, a very small number of studies used unsupervised learning techniques, such as K-means clustering, to model data distributions. However, starting from 2010s, with the development of generative models like GANs, research on modeling and simulating complex data distributions gradually began to emerge. Particularly, research focused on simulating data distributions places a primary emphasis on addressing the "non-ideal distribution problem", which is one of the challenges in the tabular data generation domain.

In 2017, Srivastava et al. [56] conducted pioneering research in table data generation by applying GANs for the first time. They proposed a model for simulating engineering data distributions without labels and constructed a reconstructor network to address the persistent issue of mode collapse in GANs. Notably, this research focused on solving the mode collapse problem without considering the distribution or characteristics of table data. In 2018, Xu et al. [42] improved the quality of generated data by inserting a pre-trained autoencoder (AE) between the generator and discriminator in a GAN. The authors used a Gaussian mixture model (GMM) to represent non-ideal distributions of tabular data. The inserted AE was then used to capture the representation of the data and compare it to that of real data. In this setup, the GAN's discriminator utilized representations derived from the AE to classify authenticity. In 2019, Xu et al. [8] proposed a conditional GAN model that not only addressed non-ideal distributions of tabular data but also tackled class imbalance issues. Specifically, they suggested preprocessing methods for continuous attributes with multi-modal distributions and imbalanced discrete attributes. Continuous attributes were normalized for multi-modal distributions using GMM, while discrete attributes were addressed using the "training-by-sampling" method, ensuring balanced sampling even with imbalanced classes. The data preprocessing methods demonstrated by Xu et al. [14] have influenced several subsequent studies on generating tabular data. In 2019, Yoon et al. [57] introduced a GAN model for generating tabular data in a time-series format. They proposed a model where the GAN's generator and discriminator have a recursive structure to generate time-dependent data. The recursive structure ensured that the generated data incorporated the influence of time. In this proposed GAN model, the generator processed data through a trainable embedding layer prior to generation. This method enabled smoother generation and discrimination in lower dimensions compared to higher dimensional original data. In 2021, Rajabi et al. [58] presented a GAN model for simulating tabular data distributions while generating fair data based on constraint conditions. Here, the authors designed a

Table 3. Evaluation results from generative models for data distribution modeling using public and custom data sets.

Type	Data set	Corr	F1-score	KS-test	KL-Div
Public	Adult	Ref. [57], GAN: 0.7730	Ref. [8], GAN: 0.6800	-	
	Census-income	-	Ref. [8], GAN: 0.5200	Ref. [52], GAN: 0.0295	
	KDD	-	-	Ref. [51], GAN: 0.2476	
	Credit	-	Ref. [8], GAN: 0.7600	-	
	Convertype	-	-	Ref. [51], GAN: 0.0192	
	Stacked-MNIST	-		-	Ref. [59], GAN: 2.9500
	Google stocks data	-		Ref. [49], GAN: 0.1020	-
	Bank marketing dataset	Ref. [57], GAN: 0.8540		-	-
	COMPAS ProPublica	Ref. [57], GAN: 0.8600		-	-
	Law-school survey	Ref. [57], GAN: 0.8470		-	-
Custom	Sines simulated data	-		Ref. [49], GAN: 0.0110	

model to synthesize fair data, preserving data privacy under constraint conditions.

Models designed to model data distributions primarily evaluate their performance based on how closely the distribution of the generated data matches that of the original data. Therefore, for this purpose, evaluation metrics such as correlation coefficients, as mentioned earlier, as well as statistical indicators like Kolmogorov-Smirnov test (KS-test) and Kullback-Leibler divergence (KL-divergence) can be used. For these metrics, higher values for KS-test and lower values for KL-divergence indicate better performance, respectively.

Research aimed at modeling distributions using generated tabular data was relatively rare prior to the emergence of high-performance generative models such as GANs, primarily since it has been considered a challenging problem compared to other objectives. Therefore, as seen in Table 3, only studies using GANs for this purpose were studied. In particular, the generative models used for this purpose focused on handling non-ideal distributions. To address these issues, various advanced forms of GANs such as Wasserstein GAN [60] and StarGAN [61] were also utilized.

3.4 Data augmentation

Nowadays, high-performance deep learning models like CTGAN [14] and β-VAE [38] have deep and complex network structures, which means there exist numerous weights and parameters to be trained. This typically requires a significant amount of training data. However, it is rare to have a large volume of training data readily available in real-world cases. To address the issue of data scarcity, research on data augmentation is actively progressing. Data augmentation methods vary depending on the type of data being handled. For example, in the case of image data, augmentation can include various operations like flipping, rotating, cropping, changing colors, and more, providing a diverse range of methods to augment the dataset. However, for tabular data, because there might be correlations and causal relationships between attributes, simple methods cannot be used for data augmentation. Therefore, methods to augment tabular data are under investigation [22, 57, 62]. Models designed exclusively for data generation, such as VAEs or GANs, are typically utilized for data augmentation problems. Nonetheless, some studies have also explored the utilization of inference models like MLPs in data augmentation.

In 2020, Cheung and Yeung [44] proposed research that utilized MLP models for the purpose of augmenting medical information. Their study introduced a triplet loss function to generate data with similar distributions even with a small amount of training data. Additionally, a discriminative loss was used to assist in training the decoder by determining the authenticity of generated data. In the same year, Ohno [63] introduced a VAE-based generative model for augmenting data required for regression model training. They also proposed a framework for creating regression models using both original and generated data. Zhang et al. [64] in 2021 proposed a Bayesian-based GAN model for augmenting tabular data. Unlike conventional MLP-based GAN models, their model featured both the generator and discriminator using Bayesian networks, leveraging probabilistic statistical parameters to generate data and perform authenticity discrimination, similar to VAEs.

Data augmentation serves the purpose of not only increasing the quantity of data for secondary tasks like classification or regression but also addressing issues related to class imbalance in the data. In 2021, Islam et al. [59] utilized data augmentation to address data imbalance issues, using VAE for data augmentation. In 2020, Koivu et al. [62] proposed a GAN-based framework for oversampling the minority class. They introduced a model that combined SMOTE-MC, one of the existing oversampling methods, with GAN, and demonstrated that this model performed well even on severely imbalanced datasets with an extreme class ratio of 2499:1. Engelmann and Lessmann [39] in 2020 also developed a GAN framework for augmenting data of the minority class to address class imbalance issues. They designed a discriminator that could uniformly sample even sparse classes and employed the Wasserstein loss function. In 2019, Shao et al. [45] proposed a GAN model for augmenting machine fault diagnosis data. The augmentation of such data was crucial as it often suffers from class

Table 4. Evaluation results from generative models for data augmentation using public and custom data sets. Note that "DL" here indicates VAE classes.

Type	Data set	F1-score	RMSE
Public	Adult	Ref. [56], GAN: 0.8510	-
	Iris	Ref. [25], DL: 0.9889	-
	Germany	Ref. [62], GAN: 0.7600	-
	Cancer	Ref. [25], DL: 0.9941	-
	Shuttle	Ref. [56], GAN: 0.9960	-
	Taiwan	Ref. [62], GAN: 0.7700	-
	Arcene	Ref. [25], DL: 0.6571	-
	Convertype	Ref. [56], GAN: 0.6910	-
	Connect	Ref. [57], GAN: 0.8030	-
Custom	Intrusion	Ref. [56], GAN: 0.9318	-
	Crash/non-crash data	Ref. [29], DL: 0.9300	-
	Ionic conductivity dataset	-	Ref. [34], DL: 0.8626
	NYC-health	Ref. [52], GAN: 0.7040	-
	Induction motor data	Ref. [63], GAN: 0.9900	-

imbalance issues. They used a 1D convolutional neural network for both the generator and discriminator models. They also added labels to the generator to induce the modeling of data distribution according to labels.

Augmented data through data generation is commonly used in secondary tasks such as classification or regression. Consequently, evaluation metrics for generative models with the purpose of data augmentation typically involve F1-score and RMSE, as shown in Table 4. Moreover, generative models designed for data augmentation often employ general example datasets, such as those available from the UCI repository [65] to evaluate their performance. This choice is because data augmentation aims to tackle specific situations such as data imbalance rather than developing models tailored to a particular domain.

Generative models aimed at data augmentation, as seen in Table 4, have been mainly based on methodologies using VAE and GAN. Based on the examined papers in this study, there exists a tendency to use VAEs when generating tabular data with fewer than 30 attributes, while GANs are preferred when dealing with datasets containing more than 30 attributes or in more complex scenarios. Therefore, it is recommended that researchers investigating generative models for such purpose choose the basic model based on the size and complexity of the given problem.

4. Limitations and challenges

Despite the significant advancements in generative models for tabular data, there are still many challenges and issues due to the unique characteristics of the tabular data and fundamental problems inherent to generative models.

In this section, we highlight some of the major limitations and challenges that are central to the research on generative models for tabular data. We anticipate that future research endeavors will continue to address these issues and bring about corresponding solutions.

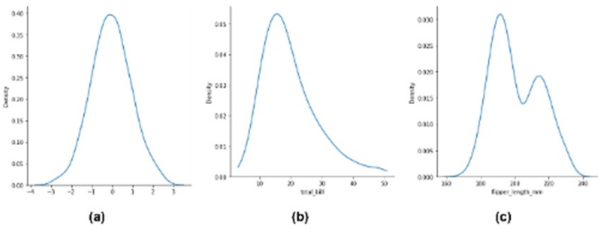


Fig. 12. Distribution types: (a) normal distribution; (b) normal distribution skewed to one side; (c) distribution with two modes. (b) and (c) belong to abnormal distribution.

4.1 Abnormal distribution

Most idealized generative models are designed with the assumption that the original data follows a normal distribution, as shown in Fig. 12(a). In reality, however, collected data often deviates from the normal distribution. It can exhibit a skewed long-tail distribution as shown in Fig. 12(b), or a multi-modal distribution as in Fig. 12(c). As a result, models designed with the assumption of a normal distribution can struggle to learn and replicate various non-normal data distributions that exist in the real world. While there have been efforts, as in the research by Xu et al. [14] and Tai et al. [13], to address abnormal distributions using Gaussian mixture models (GMM) as a preprocessing step, it is still challenging to model complex real-world distributions accurately.

4.2 Imbalanced category

In specific domains such as fault diagnosis or medical diagnosis, data often exhibit class imbalances as shown in Fig. 13. In these scenarios, class imbalance refers to a situation where

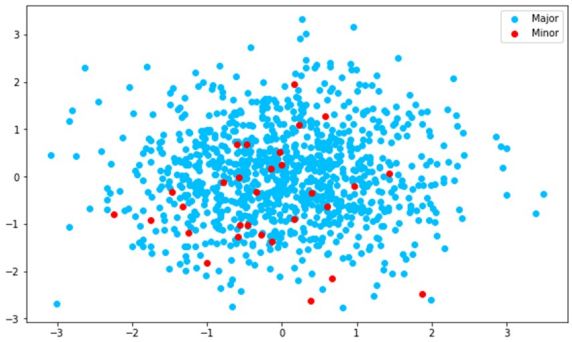


Fig. 13. Example of class imbalanced data.

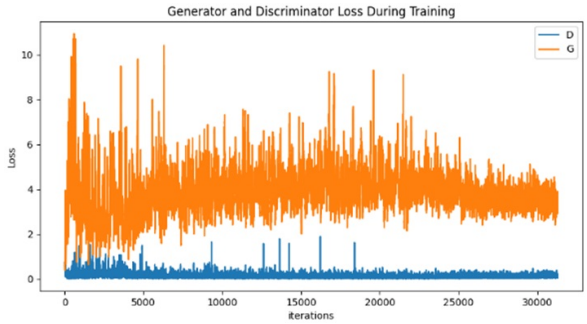


Fig. 14. Loss graph according to the learning progress of a GAN model showing an unstable learning state.

the number of data points in the minority class is significantly smaller than those in the majority class. This presents a challenge in acquiring sufficient information from the minority class. As such, the model may not be trained effectively, resulting in the generation of unrealistic data [56].

4.3 Unstable learning

Even though generative models are designed for specific purposes, their fundamental goal is to create additional data. In other words, unlike training a general inference model, these models often work with limited data. If the original data is scarce, it implies that the available information is limited. For this reason, during the training of generative models, as shown in Fig. 14, the learning process of the generator tends to be less stable compared to that of the discriminator. Therefore, the challenge of unstable learning attributed to limited data is considered a fundamentally difficult issue since it arises from the inherent problem of limited information size, making it more challenging to address compared to other problems.

4.4 Outliers

The issue of outliers is common in tabular data, as depicted in Fig. 15. When a generative model is trained with data including outliers, its performance can be degraded. Therefore, it is necessary to train generative models after removing such outliers. However, as previously explained, environments where

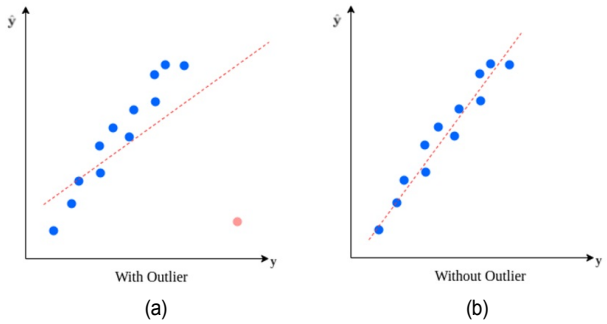


Fig. 15. Differences in model results depending on the presence: (a) or absence; (b) of outliers.

generative models are employed often involve limited amount of data. This can make it impossible to determine whether the outliers are genuine or not. Furthermore, even in scenarios where enough data is available to detect outliers, the choice of the threshold value for identifying outliers can significantly affect the performance of the generative model. Therefore, when dealing with data containing outliers, it is essential to develop methods that can robustly handle outliers before their removal.

5. Summary

This paper tried to extensively introduce research on generative models for tabular data. To assist readers new to research in this area, we introduced common terminology and notations, basic models used in tabular data generation research. We also described evaluation metrics for assessing the performance of generative models for tabular data. This paper classified various research endeavors based on their objectives and presented the current state of research for each objective along with suitable evaluation methods. In addition, we listed tables that outline the evaluation methods in each category to provide the results of performance comparisons. We also provided future research directions in each category. Finally, we discussed the limitations and challenges in this research field. We anticipate that addressing these challenges one by one will lead to significant advancements in the research field of tabular data generation.

Acknowledgments

This research was supported by Korea Institute of Marine Science & Technology Promotion (KIMST) funded by the Ministry of Oceans and Fisheries (20220210).

References

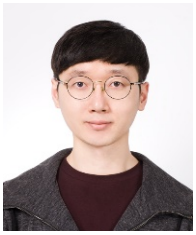
[1] J. Deng et al., Imagenet: a large-scale hierarchical image database, *2009 IEEE Conference on Computer Vision and Pattern Recognition*, Florida, USA (2009) 248-255.

[2] Z. Liu et al., *Large-scale Celebfaces Attributes (Celeba) Dataset*, Available at : https://mmlab.ie.cuhk.edu.hk/projects/CelebA.

html.
[3] A. Farhad et al., Findings of the 2021 conference on machine translation (WMT21), *Proceedings of the Sixth Conference on Machine Translation. Association for Computational Linguistics*, Online (2021).
[4] A. Maas et al., Learning word vectors for sentiment analysis, *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, Oregon, USA (2011).
[5] D. P. Kingma and M. Welling, Auto-encoding variational bayes, *arXiv:1312.6114* (2013).
[6] I. Goodfellow et al., Generative adversarial nets, *Advances in Neural Information Processing Systems 27*, Montreal, Canada (2014).
[7] C. Cortes and V. Vapnik, Support-vector networks, *Machine Learning*, 20 (1995) 273-297.
[8] J. R. Quinlan, Induction of decision trees, *Machine Learning*, 1 (1986) 81-106.
[9] L. Breiman, *Classification and Regression Trees*, Routledge, USA (2017).
[10] L. Breiman, Random forests, *Machine Learning*, 45 (2001) 5-32.
[11] A. Pinkus, Approximation theory of the MLP model in neural networks, *Acta Numerica*, 8 (1999) 143-195.
[12] M. Alzantot, S. Chakraborty and M. Srivastava, Sensegen: a deep learning architecture for synthetic sensor data generation, *2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, Hawaii, USA (2017).
[13] B.-C. Tai, S.-C. Li and Y. Huang, A VAE conversion method for private data linkage, *2021 IEEE 26th Pacific Rim International Symposium on Dependable Computing (PRDC)*, Perth, Australia (2021).
[14] L. Xu et al., Modeling tabular data using conditional gan, *Advances in Neural Information Processing Systems 32*, Vancouver, Canada (2019).
[15] J. Jordon, J. Yoon and M. V. D. Schaar, PATE-GAN: generating synthetic data with differential privacy guarantees, *International Conference on Learning Representations*, Vancouver, Canada (2018).
[16] M. K. Baowaly et al., Synthesizing electronic health records using improved generative adversarial networks, *Journal of the American Medical Informatics Association*, 26 (3) (2019) 228-241.
[17] J. P. Reiter, Using CART to generate partially synthetic public use microdata, *Journal of Official Statistics*, 21 (3) (2005) 441.
[18] G. Caiola and J. P. Reiter, Random forests for generating partially synthetic, categorical data, *Trans. Data Priv.*, 3 (1) (2010) 27-42.
[19] Y. Sun, A. Cuesta-Infante and K. Veeramachaneni, Learning vine copula models for synthetic data generation, *Proceedings of the AAAI Conference on Artificial Intelligence*, Hawaii, USA, 33 (1) (2019).
[20] H. Chen et al., FakeTables: Using GANs to generate functional dependency preserving tables with bounded real data, *Proceedings of the 28th IJCAI*, Macao, China (2019).
[21] S.-C. Li, B.-C. Tai and Y. Huang, Evaluating variational autoencoder as a private data release mechanism for tabular data, *2019 IEEE 24th Pacific Rim International Symposium on Dependable Computing (PRDC)*, Kyoto, Japan (2019).
[22] E. Choi et al., Generating multi-label discrete patient records using generative adversarial networks, *Proceedings of the 2nd Machine Learning for Healthcare Conference*, Massachusetts, USA (2017).
[23] C. Frogner et al., Learning with a Wasserstein loss, *Advances in Neural Information Processing Systems 28*, Montreal, Canada (2015).
[24] H. Han, W.-Y. Wang and B.-H. Mao, Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning, *Advances in Intelligent Computing. ICIC 2005. Lecture Notes in Computer Science,*Springer, Berlin, Heidelberg, 3644 (2005).
[25] D. Kaur et al., Application of Bayesian networks to generate synthetic health data, *Journal of the American Medical Informatics Association*, 28 (4) (2021) 801-811.
[26] A. Goncalves et al., Generation and evaluation of synthetic patient data, *BMC Medical Research Methodology*, 20 (1) (2020) 1-40.
[27] J. Drechsler, Using support vector machines for generating synthetic datasets, *Privacy in Statistical Databases. PSD 2010. Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg, 6344 (2010).
[28] B.-C. Tai et al., Exploring the relationship between dimensionality reduction and private data release, *2018 IEEE 23rd Pacific Rim International Symposium on Dependable Computing (PRDC)*, Taipei, Taiwan (2018).
[29] N. Park et al., Data synthesis based on generative adversarial networks, *arXiv:1806.03384* (2018).
[30] V. Dumoulin and F. Visin, A guide to convolution arithmetic for deep learning, *arXiv:1603.07285* (2016).
[31] M. S. Walia, B. Tierney and S. McKeever, Synthesising tabular datasets using Wasserstein conditional GANs with gradient penalty (WCGAN-GP), *28th Irish Conference on Artificial Intelligence and Cognitive Science*, Dubline Ireland, Ireland (2020).
[32] Z. Zhao et al., CTAB-GAN: effective table data synthesizing, *Asian Conference on Machine Learning, PMLR*, Online (2021).
[33] D. A. Reynolds, Gaussian mixture models, *Encyclopedia of Biometrics*, Springer (2009) 659-663.
[34] J. Zhang et al., Privbayes: private data release via Bayesian networks, *ACM Transactions on Database Systems (TODS)*, 42 (4) (2017) 1-41.
[35] J. H. Lee, I. Y. Kim and C. M. O'Keefe, On regression-treebased synthetic data methods for business data, *Journal of Privacy and Confidentiality*, 5 (1) (2013).
[36] A. Mottini, A. Lheritier and R. Acuna-Agost, Airline passenger name record generation using generative adversarial networks, *arXiv:1807.06657* (2018).
[37] M. G. Bellemare et al., The Cramer distance as a solution to biased Wasserstein gradients, *arXiv:1705.10743* (2017).
[38] H. Akrami et al., Robust variational autoencoder for tabular data with beta divergence, *arXiv:2006.08204* (2020).

[39] J. Engelmann and S. Lessmann, Conditional wasserstein GAN-based oversampling of tabular data for imbalanced learning, *Expert Systems with Applications*, 174 (2021) 114582.
[40] H. C. Valdiviezo and S. V. Aelst, Tree-based prediction on incomplete data using imputation or surrogate decisions, *Information Sciences*, 311 (2015) 163-181.
[41] J. Xia et al., Adjusted weight voting algorithm for random forests in handling missing values, *Pattern Recognition*, 69 (2017) 52-60.
[42] L. Xu and K. Veeramachaneni, Synthesizing tabular data using generative adversarial networks, *arXiv:1811.11264* (2018).
[43] S. Rashidian et al., SMOOTH-GAN: towards sharp and smooth synthetic EHR data generation, *Artificial Intelligence in Medicine: 18th International Conference on Artificial Intelligence in Medicine, AIME 2020*, Minneapolis, USA (2020).
[44] T.-H. Cheung and D.-Y. Yeung, Modals: modality-agnostic automated data augmentation in the latent space, *International Conference on Learning Representations*, Online (2020).
[45] S. Shao, P. Wang and R. Yan, Generative adversarial networks for data augmentation in machine fault diagnosis, *Computers in Industry*, 106 (2019) 85-93.
[46] O. Hummel et al., A collection of software engineering challenges for big data system development, *2018 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, Prague, Czech (2018).
[47] G. Boquet et al., Missing data in traffic estimation: A variational autoencoder imputation method, *ICASSP 2019- 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brighton, UK (2019).
[48] L. F. Burgette and J. P. Reiter, Multiple imputation for missing data via sequential regression trees, *American Journal of Epidemiology*, 172 (9) (2010) 1070-1076.
[49] L. L. Doove, S. V. Buuren and E. Dusseldorp, Recursive partitioning for missing data imputation in the presence of interaction effects, *Computational Statistics & Data Analysis*, 72 (2014) 92-104.
[50] J. T. McCoy, S. Kroon and L. Auret, Variational autoencoders for missing data imputation with application to a simulated milling circuit, *IFAC-PapersOnLine*, 51 (21) (2018) 141-146.
[51] X. Yan et al., Missing value imputation based on Gaussian mixture model for the internet of things, *Mathematical Problems in Engineering* (2015).
[52] Y. Zhao and M. Udell, Missing value imputation for mixed data via gaussian copula, *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Online (2020).
[53] T. Bedford and R. M. Cooke, Vines--a new graphical model for dependent random variables, *The Annals of Statistics*, 30 (4) (2002) 1031-1068.
[54] J. Gao et al., A survey on deep learning for multimodal data fusion, *Neural Computation*, 32 (5) (2020) 829-864.
[55] Z. Yuxuan and M. Udell, Missing value imputation for mixed data via Gaussian copula, *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Online (2020).
[56] A. Srivastava et al., Veegan: reducing mode collapse in gans using implicit variational learning, *Advances in Neural Information Processing Systems 30*, California, USA (2017).
[57] J. Yoon, D. Jarrett and M. Van der Schaar, Time-series generative adversarial networks, *Advances in Neural Information Processing Systems 32*, Vancouver, Canada (2019).
[58] A. Rajabi and O. O. Garibay, Tabfairgan: fair tabular data generation with generative adversarial networks, *Machine Learning and Knowledge Extraction*, 4 (2) (2022) 488-501.
[59] Z. Islam et al., Crash data augmentation using variational autoencoder, *Accident Analysis & Prevention*, 151 (2021) 105950.
[60] M. Arjovsky, S. Chintala and L. Bottou, Wasserstein generative adversarial networks, *Proceedings of the 34th International Conference on Machine Learning*, Sydney, Australia (2017).
[61] Y. Choi et al., Stargan: Unified generative adversarial networks for multi-domain image-to-image translation, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* Utah, USA (2018).
[62] A. Koivu et al., Synthetic minority oversampling of vital statistics data with generative adversarial networks, *Journal of the American Medical Informatics Association*, 27 (11) (2020) 1667-1674.
[63] H. Ohno, Auto-encoder-based generative models for data augmentation on regression problems, *Soft Computing*, 24 (11) (2020) 7999-8009.
[64] Y. Zhang et al., GANBLR: a tabular data generation model, *2021 IEEE International Conference on Data Mining (ICDM)*, Auckland, New Zealand (2021).
[65] A. Asuncion and D. Newman, *UCI Machine Learning Repository*, University of California, USA (2007).

Dong-Keon Kim received the B.S. degree in systems management engineering from the School of Electronic and Electrical Engineering, Sungkyunkwan University, in 2020, and the M.S. degree from the Department of Software, Sungkyunkwan University, in 2022. He is now a senior research engineer in PIDOTECH Inc. since 2022. His research interests include computer vision, machine learning, and time series forecasting.



Dongheum Ryu received a B.S. degree in Mechanical Engineering from Hanyang University, Seoul, South Korea in 2013. He then received a Ph.D. degree in Mechanical Convergence Engineering from Hanyang University, Seoul, South Korea in 2018. He is now a senior research engineer in PIDOTECH Inc. since 2018. His research interests include deep learning, generative models, machine learning, metamodeling, design of experiments, design optimization.

Yongbin Lee received a B.S. degree in Mechanical Engineering from Hanyang University, Seoul, South Korea, in 2002. He then received a M.S. degree in Mechanical Engineering from Hanyang University, Seoul, South Korea, in 2004. He then received a Ph.D. degree in Mechanical Engineering from Hanyang University, Seoul, South Korea, in 2009. He is now a senior research engineer in PIDOTECH since 2012. His research interests include machine learning, metamodeling, design of experiments, and design optimization.

Dong-Hoon Choi graduated from Seoul National University with a B.S. in mechanical engineering in 1975. He then graduated from KAIST in 1977 with an M.S. in Mechanical Engineering. He earned his Ph.D. in mechanical engineering from the University of Wisconsin-Madison. From 1986 to 2018, he served as a Professor of mechanical engineering at Hanyang University. He has been the CEO of PIDOTECH Inc. since 2003. His research interest includes AI-aided design optimization, multidisciplinary design optimization, surrogate-based design optimization, and AI applications for simulation and engineering design.