Generative AI for Synthetic Tabular Data: Privacy, Utility, and Sensitivity of Downstream Policy Inference using HRS Data*

Lakshmi K. Raut[1]

[1]Visiting Fellow, Universityof Chicago

[1]lakshmiraut@gmail.com

2025-10-01

Abstract

Strict privacy regulations restrict the public release of micro-level data critical for modeling in healthcare, education, economics, and finance. Synthetic tabular data provides a promising alternative by replicating statistical properties of the original datasets while safeguarding individual identities. This paper investigates the sensitivity of econometric policy models to the substitution of real data with synthetic data.We present an overview of differential privacy frameworks and DP-SGD algorithms that ensure formal privacy guarantees. In addition, we examine generative approaches—including GANs, VAEs, and diffusion models—with particular emphasis on Denoising Diffusion Probabilistic Models (DDPMs). We highlight the continuous-time stochastic differential equation formulation that unifies discrete diffusion processes, where the Fokker–Planck equation offers a principled simplification of backward denoising dynamics.We propose the Mahalanobis D^2 statistic as a novel metric for measuring policy sensitivity to data substitution. Using Health and Retirement Study (HRS) data, we train and assess 13 tabular generative models (three GAN-based, two VAE-based, five diffusion-based, and three additional architectures). Models are ranked across utility, privacy, and Mahalanobis D^2 metrics, providing a comprehensive benchmark for synthetic data generation in econometric policy analysis.

*This research was conducted without external grant funding and was completed independently during the author's personal time.

Contents

1 Introduction

Micro-level data is essential for developing robust statistical and machine learning models in sectors like healthcare, economics, education, and finance. Although agencies collect high-value data through services and surveys, disseminating this information for evidence-based analysis is hindered by privacy mandates, such as the Health Insurance Portability and Accountability Act (HIPAA) in the United States, the General Data Protection Regulation (GDPR) in Europe, and sector-specific frameworks like the Gramm-Leach-Bliley Act for financial data. Consequently, synthetic tabular data is gaining traction as a viable alternative that mitigates privacy risks while preserving data utility, ensuring that the statistical distributions of the synthetic data closely mirror those of the ground truth.

Beyond privacy, generative models for tabular data are increasingly important due to their multifaceted applications, which include imputing missing values, balancing minority class data for statistical analysis, and augmenting training data. The issue of *class imbalance* is common in domains like fraud detection and rare disease diagnosis. Models trained on such skewed distributions exhibit a strong bias toward the majority class, resulting in poor predictive accuracy for the critical minority class. Generative models can be conditioned to oversample the minority class, providing a rich, diverse set of new examples and enabling the creation of perfectly balanced datasets. This directly addresses the class imbalance problem in a far more sophisticated manner than traditional oversampling techniques like SMOTE (Synthetic Minority Over-sampling Technique) introduced in Chawla et al. (2002); for more on this, see He and Garcia (2009).

Generative tabular data models offer powerful advantages for missing value imputation by learning the underlying data distribution rather than simply estimating expectations. For instance, generative models like MIWAE (Mattei and Frellsen, 2018) handle challenging scenarios including missing-at-random mechanisms and achieve competitive accuracy while maintaining computational efficiency, making them particularly valuable for real-world applications with heterogeneous, incomplete datasets.

Training deep neural network models typically requires large volumes of data to achieve reliable results. This poses a challenge in domains such as biochemical drug discovery, where data availability is limited. To address this, Altae-Tran et al. (2017) introduced a *one-shot learning algorithm* that substantially reduces the amount of data needed to make meaningful predictions. Beyond drug discovery, synthetic data offers valuable support for semi-supervised and self-supervised learning paradigms, particularly in scenarios where labeled data is scarce or costly to obtain.

In industrial applications, synthetic tabular data accelerates model prototyping, enables robust benchmarking, and enhances reproducibility by providing standardized datasets that closely approximate real-world conditions without exposing proprietary or sensitive information. In healthcare contexts, however, the use of synthetic data requires careful consideration to avoid risks and ensure ethical application (Giuffrè and Shung, 2023; Koul et al., 2025; Mohammed et al., 2025; Vallevik et al., 2024). Compounding this challenge, real-world data often serves as a mirror to historical and societal biases, which, if left unaddressed, are learned and amplified by machine learning models, leading to discriminatory and inequitable automated decisions (Barocas and Selbst, 2016).

Traditional anonymization methods, such as k-anonymization, generalization, and suppression, often lead to a significant *trade-off between privacy and data utility*. This loss of utility can compromise downstream analytical tasks, particularly when training complex machine learning models or performing policy inference.

Generative Artificial Intelligence (AI) models have emerged as the state-of-the-art in generating synthetic data, including tabular data, offering significant improvements over classical techniques. Extensions of *Generative Adversarial Networks (GANs)* (Goodfellow et al., 2014), *Variational Autoencoders (VAEs)* (Kingma and Welling, 2013), and more recently introduced *Large Language Models (LLMs)* (W. X. Zhao et al., 2023; Matarazzo and Torlone, 2025) and *Diffusion Models (DMs)* (Sohl-Dickstein et al., 2015; Ho et al., 2020) for generation of high-quality synthetic tabular data are gaining popularity.

The generation of synthetic tabular data is not without its complexities. Unlike image or text data, tabular datasets often exhibit heterogeneous feature types (categorical, ordinal, continuous), intricate inter-feature dependencies, and domain-specific constraints. These characteristics pose unique challenges for generative modeling, necessitating specialized architectures and evaluation metrics tailored to tabular data synthesis. Some studies show that among all types of models, Diffusion Models have demonstrated superior capability in capturing complex, non-linear dependencies and multimodal distributions inherent in real-world tabular data (Capasso, 2025; Fonseca and Bacao, 2023; Zhang et al., 2024; Kotelnikov et al., 2023; Li et al., 2025; R. Shi et al., 2025; Truda, 2023).

The fundamental promise of synthetic data is to act as a high-fidelity (utility), privacy-preserving proxy for the real data by creating a dataset that contains no personally identifiable information (PII) and has no one-to-one mapping to the original records. Diverse metrics are used in the liter-

ature to assess utility and privacy when comparing a real dataset with various synthetic datasets. Going further toward guaranteeing privacy preservation, based on the rigorous differential privacy framework of Dwork (2008) and Dwork and Roth (2014), training algorithms such as the differentially private stochastic gradient descent (DP-SGD) algorithm proposed by Abadi et al. (2016) are built into the training of generative models to achieve a given level of guaranteed differential privacy.

While some papers examine the downstream machine learning efficiency of synthetic datasets (e.g., using ML-Efficiency metrics, Sajjadi et al. (2018)), few studies compare the sensitivity of policy inferences derived from the statistical parameter estimates of econometric models. This paper studies the viability of using synthetic HRS data for policy inference, assessing whether the derived conclusions—which are vital for informing public health and economic decisions—remain statistically equivalent to those drawn from the original protected data. The paper proposes the use of the Mahalanobis distance statistic D^2 (which is related to Hotelling's T^2 statistic) to assess the viability of synthetic HRS data for policy inference by testing whether derived public health and economic conclusions remain statistically equivalent to those from the original protected data.

The rest of the paper is organized as follows. In Section 2, I describe the mechanics of the generative AI models for tabular synthetic data that are based on GANs, VAEs, and Diffusion Models. I describe the concept of differential privacy (DP) and the differentially private stochastic gradient descent (DP-SGD) algorithm, which can be used in generative AI models to achieve differential privacy in the generated synthetic data. In Section 3, I describe various metrics used for the assessments of synthetic data utility, privacy level, and downstream policy sensitivity. In Section 4, I briefly describe the HRS dataset and the variables used in this paper. In Section 5, I compute the utility and privacy metrics for all the models to benchmark synthetic datasets generated by the 13 models considered in this paper and discuss the recommendation of models based on the utility and privacy metrics. In Section 6, I compute the policy sensitivity metric, the Mahalanobis D^2 proposed in this paper for all 13 models and discuss the recommendations for the models using this metric. Section 7 concludes the paper.

2 Generative models for synthetic tabular data

This paper aims to investigate the trade-off between privacy preservation and data utility in synthetic data generation models, and to assess the sensitivity of policy analysis using synthetic datasets produced by various machine learning models. We begin by explaining the concept of

differential privacy.

2.1 Differential Privacy (DP)

Cynthia Dwork and colleagues (Dwork, 2008) introduced a rigorous mathematical framework for the concept of privacy called *Differential Privacy (DP)*, designed to protect individual data contributions when performing statistical analysis or machine learning. The core idea is that the output of an algorithm should not significantly change whether or not any single individual's data is included, thereby ensuring plausible deniability for participants.

In short, adding or removing a single user should not statistically change the output. This stability guarantees that the model does not memorize or reveal sensitive information about any specific individual

Consider an algorithm that acts on some dataset and produces some output such as a synthetic dataset, or mean, median, mode of a variable in the real dataset. An algorithm can be a database query producing outputs of the above types. In our context, an algorithm is a machine learning model that acts on real data and produces a synthetic dataset similar in nature of the real data. Let $\mathcal{D}$ be the set of all datasets and $\mathcal{R}$ be the set of all possible outcomes of the algorithms.

A randomized algorithm is said to be *differentially private* if its outputs are nearly indistinguishable when run on two datasets in $\mathcal{D}$ that differ by only one individual's record. This is typically achieved by carefully adding noise to computations, balancing privacy guarantees with utility.

A randomized algorithm $\mathcal{M} : \mathcal{D} \to \mathcal{R}$ satisfies (ϵ, δ)-*differential privacy* if for all datasets D and D' in $\mathcal{D}$ differing in one record, and for all subsets $S \subseteq \mathcal{R}$:

$$\Pr[\mathcal{M}(D) \in S] \leq e^{\epsilon} \cdot \Pr[\mathcal{M}(D') \in S] + \delta$$

If $\delta = 0$, it is called *pure differential privacy (DP)* which provides strict guarantee and if $\delta > 0$, it called *approximate differential privacy (DP)* which allows a small probability of privacy breach.

The parameter ϵ known as *privacy budget* controls privacy loss; smaller values means stronger privacy. General practice is to treat $\epsilon < 1$ as strong privacy; $1 \leq \epsilon \leq 10$ as moderate privacy; and $\epsilon > 10$ as weak privacy. The parameter δ controls the probability of privacy breach; typically it is fixed at $\delta < \frac{1}{n^2}$ where n is dataset size.

The main mechanism for achieving (ϵ, δ)-differential privacy in Machine Learning (ML) models of synthetic data generation is to replace SGD (stochastic gradient descent) with DP-SGD (Differentially Private Stochastic Gradient Descent) in parameter estimation algorithms. The algorithm is described next.

2.2 DP-SGD (Differentially Private Stochastic Gradient Descent)

Abadi et al. (2016) developed the DP-SGD algorithm for training neural networks to achieve a level of guaranteed differential privacy, which is described below.

DP-SGD Algorithm

1. Clip gradients: For each sample gradient g_i:

$$\bar{g}_i = g_i \cdot \min\left(1, \frac{C}{\|g_i\|_2}\right)$$

where C is the clipping threshold.

2. Add noise: Compute noisy average:

$$\tilde{g} = \frac{1}{B}\left(\sum_{i=1}^{B} \bar{g}_i + \mathcal{N}(0, \sigma^2 C^2 I)\right)$$

3. Update parameters: $\theta_{t+1} = \theta_t - \eta\tilde{g}$

Privacy Accounting: Using moments accountant or Rényi DP, after T iterations:

$$\epsilon(T, \delta) = \mathcal{O}\left(\frac{q\sqrt{T\log(1/\delta)}}{\sigma}\right)$$

where $q = B/n$ is the sampling rate.

There are three main types of generative models in the literature that I use in the present study. Their mechanics are briefly described below .

2.3 GAN (Generative adversial network)

GANs, introduced by Goodfellow and colleagues (Goodfellow et al., 2014), consist of two neural networks engaged in a minimax game. The generator G maps random noise $z \sim p_z(z)$ to synthetic samples $G(z)$, while the discriminator D attempts to distinguish real samples from generated ones.

The optimization objective for training is:

$$\min_{\theta} \max_{\phi} V(D_\phi, G_\theta) = \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}}[\log D_\phi(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p_z}[\log(1 - D_\phi(G_\theta(\mathbf{z})))]$$

where p_{data} is the true data distribution and p_z is the prior distribution on latent codes (typically $\mathcal{N}(\mathbf{0}, \mathbf{I})$)

The alternating optimization procedure:

Standard GAN Training

Input: Real data $\mathcal{D}$, learning rates η_D, η_G, batch size m

1. **for** number of training iterations:
2. for k discriminator steps:
3. Sample minibatch $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}\}$ from p_{data}
4. Sample minibatch $\{\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(m)}\}$ from p_z
5. Update discriminator:

$$\phi \leftarrow \phi + \eta_D \nabla_\phi \frac{1}{m} \sum_{i=1}^{m} [\log D_\phi(\mathbf{x}^{(i)}) + \log(1 - D_\phi(G_\theta(\mathbf{z}^{(i)})))]$$

6. Sample minibatch $\{\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(m)}\}$ from p_z
7. Update generator:

$$\theta \leftarrow \theta - \eta_G \nabla_\theta \frac{1}{m} \sum_{i=1}^{m} \log(1 - D_\phi(G_\theta(\mathbf{z}^{(i)})))$$

For tabular data, this framework requires modifications to handle mixed data types (continuous, categorical, ordinal) and complex dependencies between features. Xu et al. (2019) introduced such an extension known as CTGAN which I will use this study. There are other extensions such as PAT-GAN (Jordon et al., 2018) and DP-CTGAN (Fang et al., 2022) that incorporate differential

privacy explicitly. These could not be easily adated to trained on our dataset and thus not included in the study.

2.4 VAE (Variational Auto Encoders)

A VAE model introduced by Kingma and Welling (2013) learns a probabilistic mapping between data space $\mathcal{X}$ and latent space $\mathcal{Z}$ through variational inference. Unlike GANs, VAEs have an explicit probabilistic framework and optimize a principled objective function (the evidence lower bound).

The VAE defines a generative process:

$$z \sim p_\theta(z) = \mathcal{N}(0, I), \qquad x|z \sim p_\theta(x|z) \tag{1}$$

The goal is to maximize the marginal log-likelihood:

$$\log p_\theta(x) = \log \int p_\theta(x|z) p_\theta(z) dz$$

This integral is intractable for complex decoders $p_\theta(x|z)$. Introduce an approximate posterior (encoder) $q_\phi(z|x)$ and apply Jensen's inequality:

$$\begin{aligned}
\log p_\theta(x) &= \log \int p_\theta(x, z) dz \\
&= \log \int q_\phi(z|x) \frac{p_\theta(x, z)}{q_\phi(z|x)} dz \\
&\geq \int q_\phi(z|x) \log \frac{p_\theta(x, z)}{q_\phi(z|x)} dz \\
&= \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x, z) - \log q_\phi(z|x)] \\
&= \mathcal{L}(\theta, \phi; x)
\end{aligned}$$

$\mathcal{L}(\theta, \phi; x)$ is known as *the Evidence Lower Bound (ELBO)*.

The gap between ELBO and log-likelihood is:

$$\log p_\theta(x) - \mathcal{L}(\theta, \phi; x) = D_{KL}(q_\phi(z|x) \| p_\theta(z|x))$$

Since $D_{KL} \geq 0$, maximizing the ELBO provides a lower bound on the log-likelihood and minimizes the KL divergence to the true posterior.

Reparameterization Trick

To enable backpropagation through stochastic nodes, reparameterize:

$$z = \mu_\phi(x) + \sigma_\phi(x) \odot \epsilon, \quad \epsilon \sim \mathcal{N}(0, I)$$

This separates the stochasticity (ϵ) from the parameters (ϕ), allowing gradients:

$$\nabla_\phi \mathbb{E}_{q_\phi(z|x)}[f(z)] = \mathbb{E}_{\epsilon \sim \mathcal{N}(0,I)}[\nabla_\phi f(\mu_\phi(x) + \sigma_\phi(x) \odot \epsilon)]$$

The training objective maximizes the Evidence Lower Bound (ELBO):

$$\mathcal{L}(\theta, \phi; x) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - D_{KL}(q_\phi(z|x) \| p(z))$$

The first term encourages reconstruction accuracy, while the second regularizes the latent space to match a prior $p(z) = \mathcal{N}(0, I)$.

For tabular data with mixed categorical, numeric and oridinal data, the synthetic data vault group introduced TVAE, an extension of the above VAE. Another extension of the vanilla VAE for tabular data is TTVAE by A. X. Wang and Nguyen (2025), an attention based transformer model.

VAEs offer inherent privacy benefits: (1) The KL divergence term creates a continuous, smooth latent representation, reducing memorization. (2) The probabilistic encoder adds noise during training, providing a form of implicit privacy protection. (3) The ELBO objective is more amenable to differential privacy mechanisms than GAN objectives.

Weggenmann et al. (2022) introduce the DP-VAE model that incorporates explicitly differential privacy. But their codes could not be readily adapted to our dataset and thus not included in this study.

I will include TVAE and TTVAE in this study.

2.5 Diffusion Models for Tabular Data

Diffusion models, particularly Denoising Diffusion Probabilistic Models (DDPMs), have emerged as powerful generative models. They define a forward process that gradually adds noise to data and learn a reverse process that removes noise to generate samples. In what follows, I will provide a terse presentation of this method. The details could be found in the original papers (Sohl-Dickstein et al., 2015; Ho et al., 2020). For more friendly expositions, see (Luo, 2022; Chan, 2024).

The foundational concept introduced by Sohl-Dickstein et al. was to replace a single-step conversion in VAE with a chain of sequential conversions. Specifically, they defined two distinct processes $x_0, x_1, \ldots x_T$, each x_i is in data space. One process is called *forward process* (i.e., going forward in time starting at $t = 0$) with joint distribution $q_\phi(x_{0:T})$, mirroring the encoder component. The other is a *backward process* (i.e., going backward in time starting at $t = T$) with joint distribution $p_\theta(x_{0:T})$ mirroring the decoder component of a Variational Autoencoder (VAE).

To ensure both tractability and flexibility, a Markov chain structure is imposed on these processes. This means that each state in the sequence depends only on the immediately preceding state:

$$q_\phi(x_{1:T}|x_0) = q(x_0) \prod_{t=1}^{T} q_\phi(x_t|x_{t-1}) \tag{2}$$

and

$$p_\theta(x_{0:T}) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t) \tag{3}$$

where $q(x_0)$ is the unknown data distribution that we are trying to approximate, $p(x_T)$ is a **known** distribution, generally assumed to be standard normal distribution, from which one can easily draw samples; the conditional distributions $q_\phi(x_t|x_{t-1})$ and $p_\theta(x_{t-1}|x_t)$ represent the single-step transitions of the forward and reverse processes, respectively. The parameters ϕ and θ characterize distributions of each process. The forward process is a **fixed** (specified by the user) Markov chain that adds Gaussian noise over T timesteps:

$$q_\phi(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I) \tag{4}$$

where $\beta_1, \ldots, \beta_T$ is a variance schedule with $0 < \beta_t < 1$. While other distributions could be assumed for the transition probabilities, there are advantages if these are taken to be normal.

Consequently, one can sample x_t directly given x_0 using the following,

$$q_\phi(x_t|x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)I) \tag{5}$$

where $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$. To see how Eq. (3) is derived, notice that using the reparameterization $x_t = \sqrt{\alpha_t}x_{t-1} + \sqrt{1-\alpha_t}\epsilon_{t-1}$ repeatedly, one gets,

$$\begin{aligned} x_t &= \sqrt{\alpha_t}x_{t-1} + \sqrt{1-\alpha_t}\epsilon_{t-1} \\ &= \sqrt{\alpha_t}(\sqrt{\alpha_{t-1}}x_{t-2} + \sqrt{1-\alpha_{t-1}}\epsilon_{t-2}) + \sqrt{1-\alpha_t}\epsilon_{t-1} \\ &= \sqrt{\alpha_t\alpha_{t-1}}x_{t-2} + \sqrt{\alpha_t(1-\alpha_{t-1})}\epsilon_{t-2} + \sqrt{1-\alpha_t}\epsilon_{t-1} \end{aligned}$$

Since the sum of independent Gaussians is Gaussian with variance sum, this simplifies to the closed form above.

The reverse process learns to denoise:

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \tag{6}$$

The joint distribution is:

$$p_\theta(x_{0:T}) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t)$$

where $p(x_T) = \mathcal{N}(x_T; 0, I)$.

Training Objective: The training maximizes the ELBO:

$$\mathbb{E}_q[\log p_\theta(x_0)] \geq \mathbb{E}_q\left[\log \frac{p_\theta(x_{0:T})}{q(x_{1:T}|x_0)}\right] = \mathcal{L}$$

This decomposes into:

$$\mathcal{L} = \mathbb{E}_q\left[\underbrace{D_{KL}(q(x_T|x_0)\|p(x_T))}_{L_T} + \sum_{t>1}\underbrace{D_{KL}(q(x_{t-1}|x_t,x_0)\|p_\theta(x_{t-1}|x_t))}_{L_{t-1}} - \underbrace{\log p_\theta(x_0|x_1)}_{L_0}\right]$$

Posterior $q(x_{t-1}|x_t, x_0)$: By Bayes' theorem:

$$q(x_{t-1}|x_t, x_0) = \mathcal{N}(x_{t-1}; \tilde{\mu}_t(x_t, x_0), \tilde{\beta}_t I)$$

where:

$$\tilde{\mu}_t(x_t, x_0) = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}x_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}x_t$$

$$\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$$

Simplified Training Loss: Using the connection between x_0 and x_t:

$$x_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1-\bar{\alpha}_t}\epsilon)$$

where $\epsilon \sim \mathcal{N}(0, I)$ is the noise added in the forward process.

The model can predict $\epsilon_\theta(x_t, t)$, leading to the simplified loss:

$$L_{simple} = \mathbb{E}_{t,x_0,\epsilon}\left[\|\epsilon - \epsilon_\theta(x_t, t)\|^2\right]$$

where $t \sim \text{Uniform}(1, T)$ and $\epsilon \sim \mathcal{N}(0, I)$.

DDPM Sampling Algorithm

1. Sample $x_T \sim \mathcal{N}(0, I)$ 2. For $t = T, \ldots, 1$:

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta(x_t, t) \right) + \sqrt{\beta_t} z$$

where $z \sim \mathcal{N}(0, I)$ if $t > 1$, else $z = 0$.

Score-Based Perspective: From Eq. (5), it can be seen that diffusion models connect to score matching through:

$$\nabla_{x_t} \log q(x_t) = -\frac{1}{\sqrt{1 - \bar{\alpha}_t}} \epsilon$$

The model learns the score function:

$$s_\theta(x_t, t) = -\frac{1}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta(x_t, t) \approx \nabla_{x_t} \log q(x_t)$$

This enables sampling via Langevin dynamics and connects discrete-time diffusion to continuous-time stochastic differential equations (SDEs). This leads to Flow matching (Lipman et al., 2023; Albergo and Vanden-Eijnden, 2022), an alternative to the above method, by learning continuous normalizing flows. Both frameworks above can be formulated in terms of probability flow ODE (Song et al., 2021). I describe it in a more unified framework in the next subsection.

2.6 Flow matching and diffusion models in continuous time — a unified modern approach

The fundamental insight underlying these approaches is elegant: creating noise from data is trivial, but the reverse process—generating data from noise—constitutes the essence of generative modeling. This transformation is accomplished by constructing probability paths that smoothly interpolate between a simple prior distribution (typically a Gaussian noise) and the unknown data distribution. The goal is using a neural networks to learn the vector fields or score functions that guide this transformation.

This subsection closely follows Holderrieth and Erives (2025). Both Flow matching models and diffusion models rely on differential equations—ordinary differential equations (ODEs) for flow models and stochastic differential equations (SDEs) for diffusion models—to gradually transform simple noise distributions into complex data distributions.

Consider a data distribution p_{data} from which we wish to sample. The generative process begins by defining a *probability path* $\{p_t\}_{t\in[0,1]}$, where p_0 represents a simple noise distribution (e.g., standard Gaussian) and $p_1 = p_{\text{data}}$ is the target data distribution. This path describes how probability mass evolves from noise to data over the time interval $[0, 1]$.

For flow models, the evolution of samples along this path is governed by an ODE:

$$\frac{dx_t}{dt} = u_t(x_t)$$

where $u_t : \mathbb{R}^d \to \mathbb{R}^d$ is the *vector field* at time t. The vector field determines how individual samples flow through space to transform the initial distribution p_0 into p_1.

For diffusion models, the process includes stochastic components and is described by an SDE:

$$dx_t = u_t(x_t)dt + g_t dW_t$$

where W_t denotes Brownian motion, f_t is the drift term, and g_t controls diffusion intensity. The stochasticity enables more flexible probability transformations.

The **Fokker-Planck equation** is the cornerstone connecting SDEs to probability density evolution. It describes how the probability density $p_t(x)$ evolves when particles follow an SDE. For the general SDE above, the Fokker-Planck equation states:

$$\frac{\partial p_t(x)}{\partial t} = -\nabla \cdot (u_t(x)p_t(x)) + \frac{1}{2}g_t^2 \Delta p_t(x)$$

where ∇ denotes divergence and Δ is the Laplacian operator. The first term captures transport due to the drift, while the second term models diffusion.

For ODEs (when $g_t = 0$), this reduces to the *continuity equation*:

$$\frac{\partial p_t(x)}{\partial t} = -\nabla \cdot (u_t(x) p_t(x))$$

The Fokker-Planck equation is fundamental because it provides the theoretical guarantee: if we correctly parameterize our vector field or score function, samples generated by solving the differential equation will have the desired marginal distributions at each time t.

2.6.1 Flow Matching (using differential equations)

Flow matching trains continuous normalizing flows by regressing onto conditional vector fields rather than computing expensive maximum likelihood objectives. This approach constructs simple *conditional probability paths* $p_t(x|x_1)$ that interpolate between noise p_0 and individual data samples $x_1 \sim p_{\text{data}}$.

A common choice is the Gaussian conditional path:

$$p_t(x|x_1) = \mathcal{N}(x; \mu_t(x_1), \sigma_t^2 I)$$

where μ_t interpolates from noise to data: $\mu_0 = 0$, $\mu_1 = x_1$. The *conditional vector field* for this path is:

$$u_t(x|x_1) = \frac{d\mu_t(x_1)}{dt} + \frac{1}{\sigma_t}\frac{d\sigma_t}{dt}(x - \mu_t(x_1))$$

For the simple linear interpolation $\mu_t(x_1) = tx_1$ and $\sigma_t = 1 - t$, this becomes:

$$u_t(x|x_1) = \frac{x_1 - x}{1 - t}$$

The *marginal vector field* that governs the evolution of the entire distribution is:

$$u_t(x) = \mathbb{E}_{x_1 \sim p_{\text{data}}}[u_t(x|x_1)|x_t = x]$$

Training Algorithm:

The key insight is that minimizing the *conditional flow matching loss* $\mathcal{L}_{\text{CFM}}(\theta)$ defined below is equivalent to minimizing the marginal loss:

$$\mathcal{L}_{\text{CFM}}(\theta) = \mathbb{E}_{t\sim\mathcal{U}[0,1],x_1\sim p_{\text{data}},x_0\sim p_0}\left[\|u_\theta(t,x_t) - u_t(x_t|x_1)\|^2\right]$$

where x_t is sampled from the conditional path $p_t(\cdot|x_1)$. This loss is tractable because we can explicitly compute $u_t(x_t|x_1)$ for simple conditional paths.

The training algorithm is remarkably simple:

Flow Matching Training Algorithm

1. Sample a data point $x_1 \sim p_{\text{data}}$
2. Sample noise $x_0 \sim \mathcal{N}(0, I)$
3. Sample time $t \sim \mathcal{U}[0, 1]$
4. Compute x_t from the conditional path
5. Compute loss $\|u_\theta(t, x_t) - u_t(x_t|x_1)\|^2$
6. Update parameters via gradient descent

Sampling from Flow Models:

Sampling requires solving the learned ODE:

$$\frac{dx_t}{dt} = u_\theta(t, x_t), \quad x_0 \sim \mathcal{N}(0, I)$$

Using the Euler method with step size h:

$$x_{t+h} = x_t + h \cdot u_\theta(t, x_t)$$

More sophisticated ODE solvers (Runge-Kutta methods) provide better accuracy-efficiency trade-offs.

2.6.2 Diffusion models (using stochastic differential equations)

Score-Based Formulation Diffusion models learn to reverse a forward noising process. The forward SDE gradually adds noise:

$$dx_t = f_t x_t dt + g_t dW_t$$

The reverse-time SDE that transforms noise back to data is:

$$dx_t = \left[f_t x_t - g_t^2 \nabla \log p_t(x_t)\right] dt + g_t d\bar{W}_t$$

where $\nabla \log p_t(x)$ is the score function and $\bar{W}_t$ is a reverse-time Brownian motion.

Denoising Score Matching Similar to flow matching, diffusion models employ conditional score functions. For the variance-preserving (VP) SDE with $f_t = -\frac{1}{2}\beta_t$ and $g_t = \sqrt{\beta_t}$, the conditional distribution is Gaussian:

$$p_t(x|x_1) = \mathcal{N}(x; \alpha_t x_1, \sigma_t^2 I)$$

where $\alpha_t = e^{-\frac{1}{2}\int_0^t \beta_s ds}$ and $\sigma_t^2 = 1 - \alpha_t^2$. The conditional score is:

$$\nabla \log p_t(x|x_1) = -\frac{x - \alpha_t x_1}{\sigma_t^2}$$

Training Algorithm:

The denoising score matching objective is:

$$\mathcal{L}_{\text{DSM}}(\theta) = \mathbb{E}_{t,x_1,x_0}\left[\lambda_t \|s_\theta(t, x_t) - \nabla \log p_t(x_t|x_1)\|^2\right]$$

where λ_t is a time-dependent weighting. In practice, predicting the noise ϵ rather than the score is common:

$$\mathcal{L}_\epsilon(\theta) = \mathbb{E}_{t,x_1,\epsilon}\left[\|\epsilon_\theta(t, x_t) - \epsilon\|^2\right]$$

where $x_t = \alpha_t x_1 + \sigma_t \epsilon$ with $\epsilon \sim \mathcal{N}(0, I)$.

Training procedure:

Score Matching Training Algorithm

1. Sample $x_1 \sim p_{\text{data}}$, $\epsilon \sim \mathcal{N}(0, I)$, $t \sim \mathcal{U}[0, 1]$
2. Compute $x_t = \alpha_t x_1 + \sigma_t \epsilon$
3. Predict noise: $\hat{\epsilon} = \epsilon_\theta(t, x_t)$
4. Compute loss and update: $\|\hat{\epsilon} - \epsilon\|^2$

Sampling from Diffusion Models:

The Euler-Maruyama method discretizes the reverse SDE:

$$x_{t-h} = x_t + h\left[f_t x_t - g_t^2 s_\theta(t, x_t)\right] + g_t\sqrt{h}\xi_t$$

where $\xi_t \sim \mathcal{N}(0, I)$. Deterministic sampling using the probability flow ODE is also possible:

$$x_{t-h} = x_t + h\left[f_t x_t - \frac{1}{2} g_t^2 s_\theta(t, x_t)\right]$$

2.7 Tabular Data Adaptations

Feature Preprocessing

Tabular data requires careful preprocessing and modifications in the archtectures. See, for instance, Kotelnikov et al. (2023) for the TabDDPM model:

1. **Numerical features**: Quantile transformation to approximate Gaussian distributions
2. **Categorical features**: One-hot encoding or learned embeddings
3. **Mixed representations**: Concatenated feature vectors with appropriate normalization

The architectures of each model, its training and sampling procedures can be found in the related papers. The paper Li et al. (2025) gives a comprehensive survey of various models.

Comparative Analysis

DDPMs offer: - Well-established theory and training stability - Explicit noise scheduling control - Strong performance on diverse data types

Flow matching provides: - Faster sampling via straight trajectories - Simulation-free training - Better numerical stability

Recent work suggests flow matching achieves comparable quality with 2-5× faster inference (Lipman et al., 2023).

2.8 Models of synthetic datasets studied this paper

I use the following models for the exercise of this paper for which the python codes are available to fit on our dataset. I use 13 models of various types and train them on HRS dataset and generate synthetic datasets from those models for assessing their performance.

GAN Based models:

- CTGAN (Xu et al., 2019)
- CTABGAN (Z. Zhao et al., 2021)
- CopulaGAN (Patki et al., 2021)

VAE based models:

- TVAE (Xu et al., 2019)
- TTVAE (A. X. Wang and Nguyen, 2025)

Diffusion models:

- TabDDPM (Kotelnikov et al., 2023),
- CoDi (Lee et al., 2023),
- Tabsyn (Zhang et al., 2024),
- TabDiff (J. Shi et al., 2024),
- CDTD (Mueller et al., 2023)

Other type of models:

- SMOTE (Chawla et al., 2002)
- ARF (Watson et al., 2022)
- TabularARGN (Tiwald et al., 2025)

3 Metrics for utility, privacy and policy sensitivity

Synthetic data generation practitioners have been using traditional anonymization techniques like *k-anonymity* that ensures each record is indistinguishable from at least (k-1) others with respect to quasi-identifiers like age, ZIP code and similarly *l-diversity* and few others. However, these methods have been repeatedly shown to be insufficient, as they are vulnerable to re-identification and linkage attacks, especially in high-dimensional datasets (Sweeney, 2002; Narayanan and Shmatikov, 2008). Furthermore, these methods often degrade the underlying statistical properties of the data to the point where it loses its utility for complex ML tasks. These metrics are not used in this study. The metrics used in this study are described below.

3.1 Utility

3.1.1 KL Divergence metric

The *Kullback-Leibler (KL) divergence,* $KL(P||Q)$, measures the pseudo-distance from a "true" distribution P to an "approximating" distribution Q, defined as

- For discrete distributions: $KL(P||Q) = \sum_x P(x) \log\left(\frac{P(x)}{Q(x)}\right)$
- For continuous distributions: $KL(P||Q) = \int p(x) \log\left(\frac{p(x)}{q(x)}\right) dx$

For $KL(P||Q)$ to be finite, P must be *absolutely continuous* with respect to Q. This means that anywhere P has a non-zero probability, Q must also have a non-zero probability.

Estimating KL divergence between mixed continuous and discrete distributions is theoretically complex. Standard formulas often fail because mixed distributions lack absolute continuity; specifically, comparing a discrete probability mass in P against a continuous density in Q yields infinite divergence.

Key estimation methods are:

- **Discretization:** Bins continuous variables to create fully discrete PMFs. This is computationally simple but sensitive to bin sizing and the "curse of dimensionality."

- **Monte Carlo:** Approximates the divergence expectation via sample averaging, feasible only when the specific underlying density functions are fully evaluable. More specifically, approximate this expectation by drawing many samples $x_1, \ldots, x_N$ from the distribution P and then computing the sample mean:

$$\hat{D}(P||Q) \approx \frac{1}{N} \sum_{i=1}^{N} \log \left(\frac{p(x_i)}{q(x_i)} \right)$$

  This is not possible in the present context, as the distribution function for the real data P is unknown.

- **k-NN Estimation:** A non-parametric approach using sample distances. It bypasses density estimation entirely, effectively handling high-dimensional mixed data. This method, famously proposed by Wang, Kulkarni, and Verdú (Q. Wang et al., 2009) also see (Perez-Cruz, 2008), relies on the distances between samples. For each sample x_i from distribution P, it finds the distance to its *k-th nearest neighbor* in the same set of samples from P (let's call this distance $\rho(i)$). It also finds the distance to its k-th nearest neighbor in the other set of samples from Q (let's call this $\nu(i)$). The estimator uses the ratio of these distances. A simplified version of the estimator looks like:

$$\hat{D}(P||Q) \approx \frac{d}{N} \sum_{i=1}^{N} \log \left(\frac{\nu(i)}{\rho(i)} \right) + \log \left(\frac{M}{N-1} \right),$$

  where d is the dimension of the data, N is the number of samples from P, and M is the number of samples from Q.It converges to the true KL divergence as the number of samples increases and works in high-dimensional spaces where binning fails. It gracefully handles the mix of continuous and discrete data by using a proper distance metric (e.g., Gower distance) that can accommodate both data types.

It is, however, computationally more expensive than binning and requires careful selection of the distance metric and the parameter k.

3.1.2 Propensity Mean Squared Error (pMSE)

The utility metric *pMSE (Propensity Mean Squared Error)* is a metric used to evaluate the *fidelity* (i.e., the statistical similarity) of a synthetic dataset compared to a real dataset. The core idea is to test how easily a machine learning model can "tell the difference" between a real row and a synthetic row. To compute it, one trains a classifier (like Logistic Regression or a Random Forest) to predict the "propensity" (i.e., probability) that a given row is synthetic. If the synthetic data is perfectly realistic and statistically identical to the real data, the classifier should be completely "confused." In this "confused" state, for any given row (real or synthetic), the classifier's best guess would be 0.5 (a 50/50 chance). The pMSE metric measures how far the classifier's predictions are from this ideal 0.5 value.

The formula is:

$$pMSE = \frac{1}{N}\sum_{i=1}^{N}(p_i - 0.5)^2$$

Where, N is the total number of rows (real + synthetic) and p_i is the predicted probability (propensity score) that the i-th row is synthetic.

A pMSE score near 0.0 is IDEAL. This means the classifier's predicted probabilities are all clustered around 0.5 ($p_i \approx 0.5$). The model has no idea which data is real and which is fake. This indicates high fidelity synthetic data. A pMSE score near 0.25 is POOR. This is the worst possible score. It means the classifier can perfectly separate the data. It predicts $p_i \approx 1$ for all synthetic rows (since $(1 - 0.5)^2 = 0.25$) and $p_i \approx 0$ for all real rows (since $(0 - 0.5)^2 = 0.25$). This indicates low fidelity data that is "obviously fake."

3.1.3 Nearest Neighbor Adversarial Accuracy (NNAA)

The utility metric *Nearest Neighbor Adversarial Accuracy (NNAA)* evaluates how distinguishable synthetic data is from real data by checking whether each record's nearest neighbor (in feature space) comes from the same dataset or the opposite one. If synthetic and real data are well-mixed, the classifier accuracy will be close to 50%. If they are easily separable, accuracy will be much higher, indicating poor synthetic realism.

This is how it is computed. Combine the real dataset R and synthetic dataset S. Label each record: `0` = real, `1` = synthetic. Fit a classifier such as Logistic or Random Forest classifier. For each record, find its nearest neighbor (excluding itself). Predict the label of the record as the label of its nearest neighbor. Compute the classification accuracy across all records, and compute NNAA with the

formula.

$$\text{NNAA} = \frac{\#\text{correctly predicted labels}}{\text{total records}}$$

General guidelines for NNAA is that NNAA ≈ *0.5* means synthetic and real are indistinguishable (good utility); NNAA > *0.7* indicates the datasets are too different — synthetic data lacks realism; NNAA < *0.3* Suggests mode collapse or overfitting — synthetic data may be too close to real data.

3.1.4 Kolmogorov-Smirnov statistic and Hellinger distance

For a single variable, the *Kolmogorov-Smirnov statistic* is defined as,

$$D_{KS} = \sup_x |F_{real}(x) - F_{synth}(x)|$$

where F is the empirical CDF. Lower values indicate better similarity. The scores for individual columns are aggregated to arrive at the overall metric.

The *Hellinger distance* quantifies the difference between two probability distributions $P$and Q, defined for probability mass functions as

$$H(P, Q) = \sqrt{\frac{1}{2}\sum_{i=1}^{n}\left(\sqrt{p_i} - \sqrt{q_i}\right)^2}$$

where p_i and q_i are the probabilities of the i-th outcome in distributions P and Q. Its range is $0 \leq H(P, Q) \leq 1$. A value of the distance 0 means identical distributions and a value 1 means completely disjoint distributions. A small value (close to 0) indicates synthetic data is statistically faithful.

3.1.5 Correlation Difference

The *Correlation Difference* metric is computed as follows.

$$\Delta_{corr} = \|\text{Corr}(X_{real}) - \text{Corr}(X_{synth})\|_F$$

using Frobenius norm.

3.2 Privacy metrics

3.2.1 Hit rate

The *hit rate* metric in synthetic data evaluation measures the proportion of synthetic records that exactly replicate (or nearly replicate) real records. A high hitting rate means the generator memorized and copied training data, which undermines privacy and generalization.

Given a real dataset R and a synthetic dataset S, the hitting rate is:

$$\text{hit rate} = \frac{\#\{s \in S : s \in R\}}{|S|}$$

A low hit rate close to 0 means synthetic data is not memorizing individuals. A high hitting rate close to 1 means synthetic data is leaking real records.

3.2.2 Epsilon hit rate

The *epsilon hit rate* metric measures the proportion of synthetic records that are "too close" to real records, where "too close" is defined by a user-chosen distance threshold ϵ. It is a privacy risk metric: a higher value means more synthetic records are nearly identical to real ones, increasing re-identification risk.

It is defined as

$$\text{epsilon hit rate} = \frac{1}{|S|} \sum_{x \in S} \mathbf{1}\left(\min_{y \in R} d(x, y) \leq \varepsilon \right),$$

where R is the set of real records, S is the set of synthetic records and $d(x, y)$ is the distance. High median_DCR means synthetic records are farther from real one, i.e., lower privacy risk; low median_DCRmeans synthetic records are very close to real ones, i.e., higher privacy risk.

3.2.3 Median Distance to Closest Record (median_DCR)

The *median_DCR* metric estimates how close synthetic records are to real records, helping assess the risk of re-identification. When the datasets contain categorical variables, using Euclidean distance does not make sense, one uses Gower's distance.

For each synthetic record, compute the distance to every real record. Identify the closest real

record, i.e. minimum distance of the synthetic record to all the real recods and then take the median of the shortest distances of the synthetic records. Formally,

$$\text{median\_DCR} = \text{median}\left(\min_{y \in R} d(x, y) \quad \forall x \in S\right),$$

where R is the set of real records, S is the set of synthetic records and $d(x, y)$ is the distance. High median_DCR means synthetic records are farther from real one, i.e., lower privacy risk; low median_DCR means synthetic records are very close to real ones, i.e., higher privacy risk.

This metric is often reported alongside hitting rate and epsilon identifiability risk to give a fuller privacy picture.

3.3 Policy sensitivity metric – Mahalanobis Distance

Evidence based policy analysis is sometimes based on some multivariate mean of variables of a tabular dataset, or based on statistical parameter estimates of a econometric model with a vector of parameters say θ. If the estimates of one or more policy relevant important parameters statistically significantly differ, or even worse, change signs when estimated using original data and synthetic data, the synthetic data will produce quite different policy inference as compared to the real data. I use the Mahalanobis distance D to measure the distance between $\hat{\theta}_{real}$ and $\hat{\theta}_{synth}$. Following the arguments in Johnson and Wichern (2013), Chapter 5 and Rencher and Christensen (2012), Chapter 6, and noting how the Mahalanobis D^2 is related to Hotelling's T^2 statistic for two independent samples, and its conversion to the F statistic for p-value calculation, one can test if the synthetic dataset will produce significant policy distortions or not.

Under the null hypothesis, $H_0 : \theta_{real} = \theta_{synth}$, the squared *Mahalanobis distance* D^2 is asymptotically distributed as χ^2_p (p is the dimension of the parameter vector θ).

$$D^2 = (\hat{\theta}_{real} - \hat{\theta}_{synth})' \hat{\Sigma}^{-1}_{pooled} (\hat{\theta}_{real} - \hat{\theta}_{synth}),$$

where

$$\hat{\Sigma}_{pooled} = \frac{(n_1 - 1)\hat{\Sigma}_{real} + (n_2 - 1)\hat{\Sigma}_{synth}}{n_1 + n_2 - 2},$$

where n_1 and n_2 are the sizes of real and synthetic datasets respectively.

4 The Dataset and the construction of variables

I use the Health and Retirement Study (HRS) dataset for empirical analysis. A lot has been written about HRS datasets–about its structure, purpose, and various modules collecting data on genetics, biomarkers, cognitive functioning, and more, see for instance (Juster and Suzman, 1995; Sonnega et al., 2014; Fisher and Ryan, 2017).

For definition of variables, see Raut (2024a).

The demographic variables **White** and **Female** have the standard definition. The variable **College+** is a binary variable taking value 1 if the respondent has education level of completed college and above (does not include some college), i.e., has a college degree and more and taking value 0 otherwise.

CES-D: I used the score on the Center for Epidemiologic Studies Depression (CES-D) measure in various waves that is created by RAND release of the HRS data. RAND creates the score as the sum of five negative indicators minus two positive indicators. "The negative indicators measure whether the Respondent experienced the following sentiments all or most of the time: depression, everything is an effort, sleep is restless, felt alone, felt sad, and could not get going. The positive indicators measure whether the Respondent felt happy and enjoyed life, all or most of the time." I standardize this score by subtracting 4 and dividing 8 to the RAND measure. The wave 1 had different set of questions so it was not reported in RAND HRS. I imputed it to be the first non-missing future CES-D score. In the paper, I refer the variable as CES-D. Steffick (2000) discusses its validity as a measure of stress and depression.

Cognitive scores: This variable is a measure of cognitive functioning. RAND combined the original HRS scores on cognitive function measure which includes "immediate and delayed word recall, the serial 7s test, counting backwards, naming tasks (e.g., date-naming), and vocabulary questions". Three of the original HRS cognition summary indices—two indices of scores on 20 and 40 words recall and third is score on the mental status index which is sum of scores "from counting, naming, and vocabulary tasks"—are added together to create this variable. Again due to non-compatibility with the rest of the waves, the score in the first wave was not reported in the RAND HRS. I have imputed it by taking the first future non-missing value of this variable.

HIGH BMI : The variable body-mass-index (HIGH BMI) is the standard measure used in the medical field and HRS collected data on this for all individuals. If it is missing in 1992, I impute it with the first future non-missing value for the variable. Following the criterion in the literature,

I create the variable HIGH BMI taking value 1 if HIGH BMI > 25 and value 0 otherwise.

Now I describe the construction of the behavioral variables.

Smoking: This variable is constructed to be a binary variable taking value 1 if the respondent has reported yes to ever smoked question during any of the waves as reported in the RAND HRS data and then repeated the value for all the years.

Exercising: The RAND HRS has data on whether the respondent did vigorous exercise three or more days per week. I created in each time period to be 1 if the respondent did vigorous exercise three or more days per week in any of the waves and then that value is assigned to all the years.

Childhood SES: This variable is a binary variable measuring childhood SES. I constructed it using the IRT procedure as follows. From the HRS data I created four binary variables using the original categorical data on family moved for financial reason, family usually got financial help during childhood, father unemployed during childhood, father's usual occupation during childhood (0 = disadvantaged and 1 = advantaged), and three tertiary variables two on each parent's educational levels (0 = High School dropout, 1 = some college, 2 = completed college and higher) and third on family financial situation (0 = poor, 1 = average, 2 = well-off). I used these seven variables as items in the IRT procedure to first compute a continuous score estimate and then I define **Childhood SES** = 1 if the score is above mean plus one standard deviation of the scores and 0 otherwise.

Childhood Health is a binary measure of childhood health constructed from the self-reported qualitative childhood health variable in HRS. I define **Childhood Health** = 1 if the respondent reported very good or excellent, and zero otherwise.

5 Benchmarking models with privacy and utility metrics

5.1 My own implemented metrics

Table 1: My Own computed metrics sorted by pMSE

Model	pMSE	NNAA	nnMDCR	grMDCR	eps. hit rate	Compos. score
Ref.	0.0001	0.5000	0.0000	0.0000	1.0000	0.3570
arf	0.0003	0.4823	1.4660	0.7360	0.0000	0.5758
cdtd	0.0005	0.4152	1.0200	0.7344	0.0000	0.5135
tabddp	0.0018	0.4776	1.3397	0.7346	0.0000	0.5558
smote	0.0051	0.3518	1.0033	0.8005	0.0000	0.5053
tabsyn	0.0062	0.5326	1.4434	0.7346	0.0000	0.5624
tvae	0.0120	0.6432	1.2156	0.8006	0.0000	0.5220
codi	0.0146	0.5303	1.4419	0.7490	0.0000	0.5508
tabula	0.0168	0.5331	1.4496	0.0267	0.0113	0.5315
tabdiff	0.0587	0.7564	1.7233	0.0738	0.0018	0.5125
ctabgan	0.0624	0.7816	1.7420	0.1376	0.0001	0.5142
ttvae	0.0627	0.7524	1.4511	0.0743	0.0017	0.4715
ctgan	0.0772	0.7696	1.5580	0.8011	0.0000	0.4747
copula	0.2492	0.9996	3.6659	0.1176	0.0000	0.5111

Note: pMSE = Propensity Mean Squared Error, NNAA = Nearest Neighbor Adversarial Accuracy (NNAA $\approx$ *0.5* datasets are similar), nnMDCR = nearest neighbor median distance, and gowerMDCR = Gower median distance, eps. hit rate = epsilon hitting rate (closer to *0.0* is lower privacy leaks.

The models sorted (best first) by the **privacy metrics** in Table 1 are:

- nnMDCR (nearest neighbor median distance) — **copulagan, ctabgan, tabdiff, ctgan, arf, ttvae, tabularARGN, tabsyn, codi, tabddpm, tvae, cdtd, smote**
- grMDCR (median Gower distance to closest record) — **ctgan, tvae, smote, codi, arf, tabsyn, tabddpm, cdtd, ctabgan, copulagan, ttvae, tabdiff, tabularARGN**.

5.2 Syntheval metrics

In this section, I present a selected few metrics from Lautrup et al. (2024) using their codes available on github. I have renamed some of the metrics to the names used in my own implementations.

Table 2: SynthEval utility metrics sorted by by utility rank

synth. data	pMSE	NNAA	K-S statistic	Hellinger distance	Utility rank
Ref	0.0001	0.0000	0.0000	0.0000	7.0000
arf	0.0007	0.5682	0.0648	0.0415	5.0758
tabsyn	0.0058	0.5268	0.0426	0.0257	4.7714
tabular	0.0010	0.5108	0.0172	0.0112	5.0625
tvae	0.0140	0.8660	0.1137	0.0653	3.7410
codi	0.0179	0.6632	0.1053	0.0500	3.8917
smote	0.0047	0.8024	0.1020	0.0725	4.2499
tabddp	0.0015	0.4451	0.0211	0.0152	5.3041
cdtd	0.0004	0.4441	0.0235	0.0093	5.6741
tabdiff	0.0539	1.0000	0.0931	0.0380	2.7515
ttvae	0.0581	1.0000	0.0985	0.0378	2.4446
ctabgan	0.0481	1.0000	0.1646	0.0711	1.8897
ctgan	0.0758	0.8672	0.2054	0.1128	0.9530
copula	0.2421	1.0000	0.2200	0.0748	0.3369

Notes: K-S stands for Kolmogorov-Smirnov.

The models sorted (best first) by utility metrics in Table 2 are:

- pMSE — **cdtd, arf, tabularARGN, tabddpm, smote, tabsyn, tvae, codi, ctabgan, tabdiff, ttvae, ctgan, copulagan**,
- Kolmogorov-Smirnov statistic — **tabularARGN, tabddpm, cdtd, tabsyn, arf, tabdiff, ttvae, smote, codi, tvae, ctabgan, ctgan, copulagan**,
- Hellinger distance — **cdtd, tabularARGN, tabddpm, tabsyn, ttvae, tabdiff, arf, codi, tvae, ctabgan, smote, copulagan, ctgan**.

Table 3: SynthEval privacy metrics sorted by privacy rank

synth. data	median DCR	eps_privacy loss	eps. hit rate	Privacy rank	Composite score
Ref	1.0000	-0.3963	0.0000	1.6393	8.6393
ctgan	3.6974	-0.0017	0.0833	4.8434	5.7964
tvae	3.5608	0.0274	0.1579	4.6448	8.3857
codi	3.4032	0.0480	0.2559	4.3646	8.2564
arf	4.5667	0.0616	0.3083	4.2345	9.3103
ttvae	18.1320	0.0000	0.0000	4.0000	6.4446
ctabgan	17.8736	0.0000	0.0000	4.0000	5.8897
tabdiff	16.1454	0.0000	0.0000	4.0000	6.7515
copula	15.6249	0.0000	0.0000	4.0000	4.3369
smote	3.1109	0.2064	0.3650	3.7060	7.9558
tabsyn	1.0791	0.0976	0.4608	3.6426	8.4141
tabularARGN	1.1498	0.1446	0.6023	3.3308	8.3932
tabddpm	0.6689	0.2054	0.5883	2.6483	7.9525
cdtd	0.3529	0.2909	0.6405	1.6787	7.3529

Taking into account the trade-off between utility and privacy, Lautrup et al. (2024) suggests a weighting of their metrics to come up with their composite rank metric. Table 3 reports the value of this metric for all 13 models. According to this metric, the best 5 (in descreasing order) are **arf, tabsyn, tabularARGN, tvae, codi**.

5.3 Visual comparisons

I plot performances of three synthetic datasets compared to real dataset.

Figure 1: Comparing marginal distribution real versus synthetic data by ARF

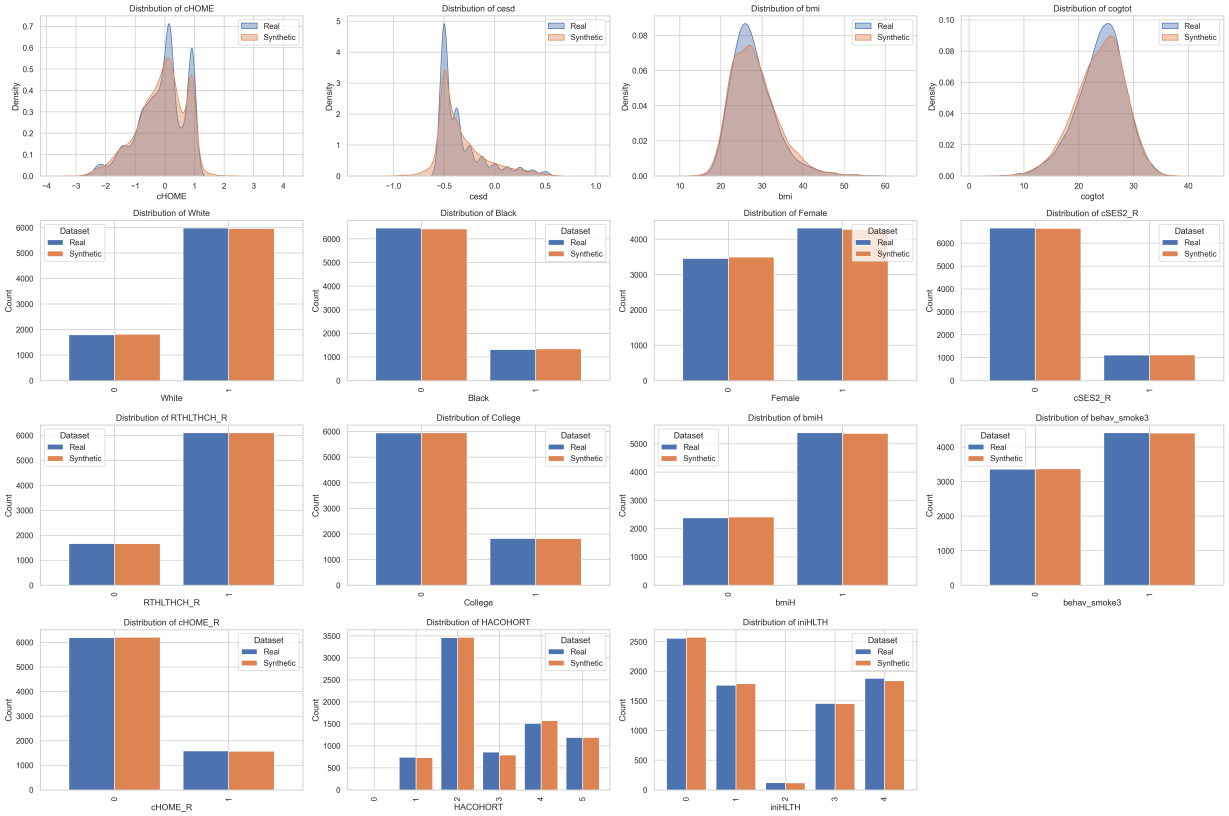
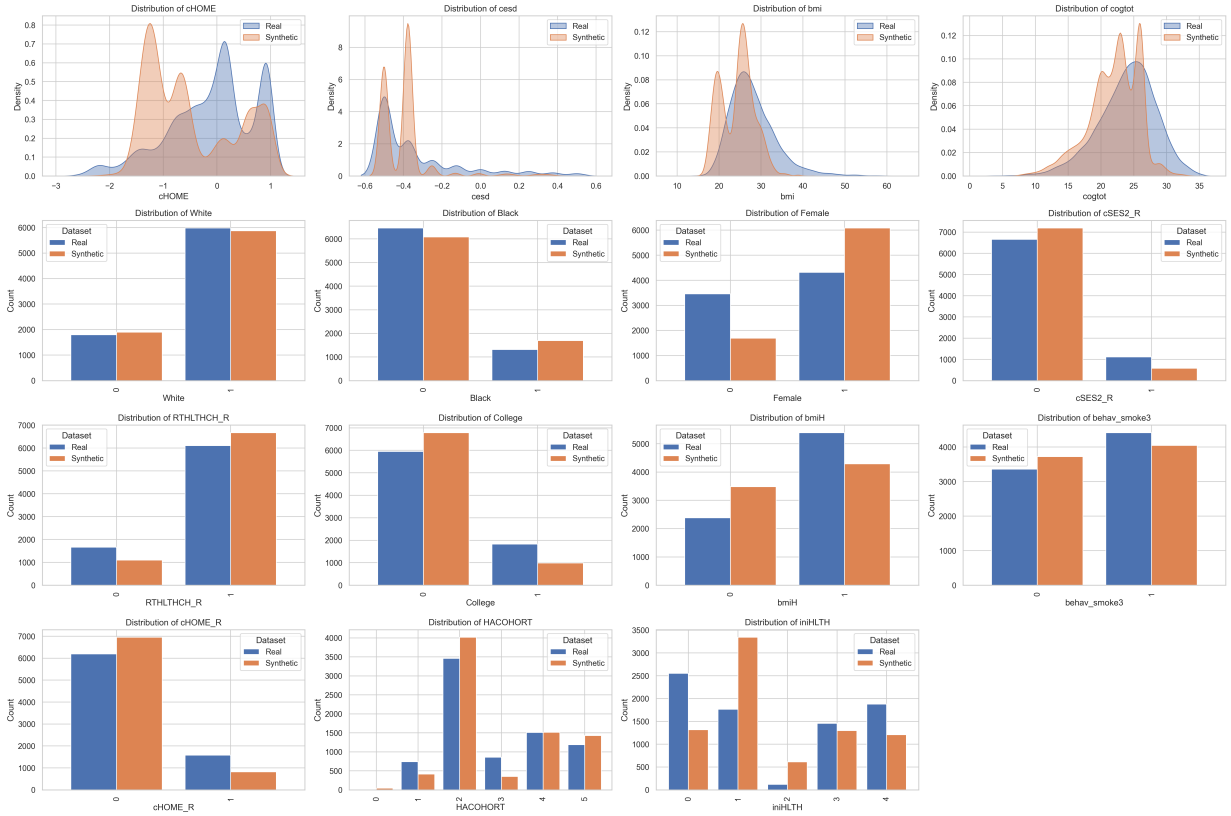
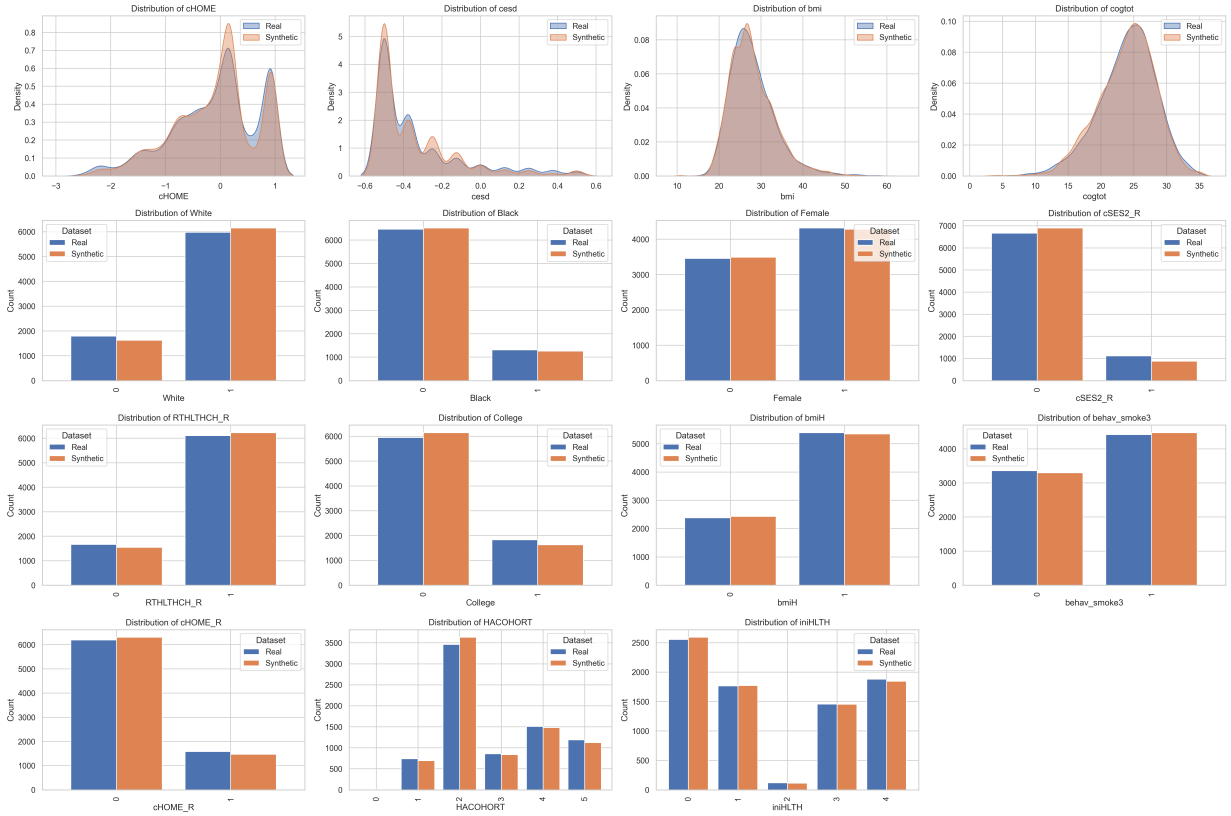


Figure 2: Comparing marginal distribution real versus synthetic data by CTGAN



I also plot the performance of the commonly used `TabDDPM` in the class of diffusion model.

Figure 3: Comparing marginal distribution real versus synthetic data by TabDDPM



6 Policy sensitivity with econometric policy models

I consider two types of policy relevant econometric models. one set studies the importance of childhood factors in the determination of childhood health, education and mid-age health (i.e., healthy or not in mid-ages). The other set studies the effect of childhood factors, health related behaviors on incidence of various chronic diseases at mid ages.

I examine the sensitivity of parameter estimates to the substitution of the real data with synthetic data, examining the differences in the significant parameters and also with the proposed measure of policy sensitivity metric, Mahalanobis distance. I highlight the models which have no statistically significant policy sensitivity. While for each set of models, I report the Mahalanobis distance metrics for all the econometric models in each set for all 13 synthetic datasets, I only present the econometric parameter estimates for two synthetic datasets – one generated by ARF which is found to produce good utility by most utility metrics; the generated by CTGAN, a widely used

method in the literature and also found above to have good value for privacy metrics.

6.1 Econometric models of childhood development

Childhood health status (Childhood Health) is an important factor for later life health outcomes and educational attainments. Childhood SES influences the stressors of the cells environment and thus will affect Childhood Health. Apart from Childhood SES, other factors such as nutrition and pediatric health care are important factors.

Many childhood factors also determine College+ such as innate IQ, family background, preschool inputs, prenatal and postnatal stressors for brain development, the childhood health status, and mother's time input. See, Heckman (2008) and Raut (2018) for recent literature on the biology of brain development and the role of socioeconomic factors, and Heckman and Raut (2016) for a Logit model of college completion in which a IQ measure, family background measured with parents' education, preschool inputs and non-cognitive skills play important roles. See Raut (2024b) for a similar model that uses the HRS dataset. The latter econometric model is used in this paper.

Table 4: Contrasting parameter estimates of regression models of childhood factors: real data versus synthetic data generated by various models

Synth data	Childhood Health	College+	Midage Health
arf	2.63	9.80	1.93
tabsyn	105.83*	114.88*	16.53*
tabularARGN	25.67*	23.97*	16.33*
tabddpm	13.72*	25.03*	2.64
codi	12.52*	1.83	9.88
tvae	438.90*	47.07*	254.69*
cdtd	6.09	5.35	9.29
tabdiff	1.49	6.50	2.62
ctabgan	2.02	8.29	22.86*
Smote	72.28*	80.76*	8.67
ttvae	7.50	16.62*	5.06
ctgan	184.14*	322.33*	461.12*
copulagan	91.11*	212.01*	177.45*

Note: A Mahalanobis distance statistic with a * means its p-value < 0.01, providing strong evidence against the null hypothesis that the parameter estimates from the real data and synthetic data are equal.

Estimates of Mahalanobis distance metric in Table 4 show that the synthetic data generators **ARF**, **CDTD**, and **TabDiff** methods are best, producing statistically identical parametric estimates for all econometric policy models of this section and the next best generators are **CoDi**, **ctabgan** and **TTVAE**, producing statistically identical for almost all models.

The parameter estimates for ARF and CTGAN are presented below to visually see the differences of statistically significant parameter estimates for the econometric models of this sub section.

Table 5: Effects of childhood factors, race and sex on childhood health and college education: real vs ARF

	cHLTH:real	cHLTH:synth	College:real	College:synth	midage health: real	midage health: synth
(Intercept)	1.089 ***	1.065 ***	-2.048 ***	-1.908 ***	-1.130 ***	-1.000 ***
	(0.065)	(0.064)	(0.097)	(0.094)	(0.078)	(0.076)
White	0.299 ***	0.349 ***	0.428 ***	0.406 ***	0.264 ***	0.170 **
	(0.063)	(0.063)	(0.074)	(0.073)	(0.060)	(0.059)
Female	-0.141 *	-0.125 *	-0.409 ***	-0.483 ***	-0.201 ***	-0.206 ***
	(0.056)	(0.056)	(0.057)	(0.056)	(0.049)	(0.049)
cSES	0.536 ***	0.342 ***	1.596 ***	1.314 ***	0.225 **	0.205 **
	(0.091)	(0.087)	(0.069)	(0.069)	(0.070)	(0.069)
RTHLTHCH_			0.544 ***	0.502 ***	0.291 ***	0.233 ***
			(0.077)	(0.075)	(0.062)	(0.061)
College					0.218 ***	0.243 ***
					(0.059)	(0.058)
N	7775	7775	7775	7775	7775	7775
R squared	0.009	0.007	0.088	0.068	0.012	0.010
Mahalanobis distance	2.629	2.629	9.802	9.802	1.933	1.933
p-value	0.453	0.453	0.044	0.044	0.858	0.858
	-	-	-	-	-	-
Loglik	4002.297	4008.616	3867.259	3941.574	4864.316	4888.015

*** $p < 0.001$; ** $p < 0.01$; * $p < 0.05$. Standard errors are in parentheses. Low p-value (e.g., < 0.01) reported below the Mahalanobis distance provides strong evidence against the null hypothesis that the parameter estimates from the real data and synthetic data are equal.

Table 6: Effects of childhood factors, race and sex on childhood health and college education - real data vs synthetic data by CTGAN

	cHLTH:real	cHLTH:synth	College:real	College:synth	midage health: real	midage health: synth
(Intercept)	1.089 ***	1.933 ***	-2.048 ***	-3.944 ***	-1.130 ***	-2.841 ***
	(0.065)	(0.095)	(0.097)	(0.204)	(0.078)	(0.137)
White	0.299 ***	-0.160 *	0.428 ***	0.936 ***	0.264 ***	0.738 ***
	(0.063)	(0.078)	(0.074)	(0.118)	(0.060)	(0.087)
Female	-0.141 *	-0.059	-0.409 ***	-0.568 ***	-0.201 ***	0.091
	(0.056)	(0.082)	(0.057)	(0.088)	(0.049)	(0.076)
cSES	0.536 ***	0.609 ***	1.596 ***	3.230 ***	0.225 **	0.828 ***
	(0.091)	(0.154)	(0.069)	(0.106)	(0.070)	(0.111)
RTHLTHCH_			0.544 ***	1.287 ***	0.291 ***	0.481 ***
			(0.077)	(0.166)	(0.062)	(0.102)
College					0.218 ***	0.482 ***
					(0.059)	(0.094)
N	7775	7775	7775	7775	7775	7775
R squared	0.009	0.004	0.088	0.244	0.012	0.043
Mahalanobis distance	184.142	184.142	322.328	322.328	461.119	461.119
p-value	0.000	0.000	0.000	0.000	0.000	0.000
Loglik	-4002.297	-3159.738	-3867.259	-2240.215	-4864.316	-3382.905

*** $p < 0.001$; ** $p < 0.01$; * $p < 0.05$. Standard errors are in parentheses. Low p-value (e.g., < 0.01) reported below the Mahalanobis distance provides strong evidence against the null hypothesis that the parameter estimates from the real data and synthetic data are equal.

6.2 Econometric models of midlife health

I consider another econometric model. This model is to see how various childhood factors and health behaviors leading up to middle ages are associated with the incidence of various chronic diseases in a multinomial logit framework. The regressors and the disease states are given as in Table 8. The detailed policy issues are discussed in more in (Raut, 2024a).

Like in the previous subsection, I first show the statistical estimates of Mahalanobis distance metric for all the datasets in Table 7 and then present detailed parameter estimates for two synthetic generative models, ARF and CTGAN.

The estimates of Mahalanobis distance metrics fin Table 7 show that the synthetic data generators **ARF**, **TabDDPM**, and **CDTD** are best, producing statistically identical parametric estimates for the econometric policy model of midlife diseases of this section and the next best generators are **tabularARGN** and **SMOTE**, producing statistically identical for almost all diseases.

Table 7: Contrasting parameter estimates of multinomial logistic regression model: real data versus synthetic data generated by various models

Synth data	2-Cardiovas	3-Cancer	4-other	5-Comorbid
arf	20.63	6.10	10.64	30.50
tabsyn	46.55*	40.84*	19.69	81.73*
tabularARGN	27.03	30.72	15.60	61.89*
tabddpm	9.76	26.83	18.97	17.87
codi	23.71	36.40*	24.09	43.96*
tvae	440.11*	2446.12*	319.36*	315.21*
cdtd	15.51	21.73	24.21	26.57
tabdiff	57.12*	17.62	42.49*	150.85*
ctabgan	76.08*	25.36	55.22*	181.97*
Smote	15.32	18.82	27.69	60.59*
ttvae	75.46*	27.29	54.45*	181.28*
ctgan	836.18*	319.12*	229.87*	234.96*
copulagan	300.88*	71.81*	166.01*	428.49*

Note: A Mahalanobis distance statistic with a * means its p-value < 0.01, providing strong evidence against the null hypothesis that the parameter estimates from the real data and synthetic data are equal.

Table 8: Contrasting parameter estimates of multinomial logistic regression model: real data versus synthetic data generated by ARF model

	2-Cardiovas		3-Cancer		4-other		5-Comorbid	
data type	**original**	**synthetic**	**original**	**synthetic**	**original**	**synthetic**	**original**	**synthetic**
(Intercept)	-0.449 *	-0.446 *	-4.536 ***	-3.431 ***	-1.090 ***	-0.721 **	-0.280	-0.059
	(0.228)	(0.217)	(0.922)	(0.671)	(0.251)	(0.234)	(0.228)	(0.216)
White	-0.096	-0.025	1.422	0.122	0.447 **	0.345 **	0.354 **	0.216
	(0.128)	(0.113)	(0.726)	(0.372)	(0.152)	(0.128)	(0.137)	(0.116)
Black	0.512 ***	0.403 **	1.224	0.218	0.239	0.049	0.590 ***	0.319 *
	(0.141)	(0.124)	(0.764)	(0.411)	(0.173)	(0.146)	(0.150)	(0.127)
Female	-0.123	0.003	0.988 ***	0.790 ***	0.590 ***	0.438 ***	0.499 ***	0.389 ***
	(0.066)	(0.065)	(0.222)	(0.211)	(0.072)	(0.070)	(0.067)	(0.066)
cSES2_R	-0.104	-0.053	-0.599	-0.351	-0.188	-0.276 **	-0.257 **	-0.184
	(0.093)	(0.090)	(0.325)	(0.300)	(0.098)	(0.099)	(0.099)	(0.096)
RTHLTHCH_R	-0.060	0.047	0.235	-0.064	-0.319 ***	-0.206 *	-0.414 ***	-0.344 ***
	(0.084)	(0.081)	(0.266)	(0.238)	(0.083)	(0.083)	(0.077)	(0.076)
College	-0.115	-0.066	0.043	0.056	0.070	-0.044	-0.152	-0.277 ***
	(0.080)	(0.077)	(0.241)	(0.233)	(0.084)	(0.082)	(0.085)	(0.083)
behav_smoke3	0.103	0.076	0.152	0.287	0.343 ***	0.192 **	0.317 ***	0.139 *
	(0.065)	(0.064)	(0.192)	(0.194)	(0.069)	(0.068)	(0.066)	(0.065)
cHOME_R	-0.012	-0.016	-0.118	-0.211	-0.215 *	-0.303 ***	-0.219 **	-0.274 ***
	(0.076)	(0.075)	(0.232)	(0.238)	(0.084)	(0.085)	(0.080)	(0.080)
bmiH	0.717 ***	0.669 ***	-0.235	-0.206	0.147 *	0.032	0.941 ***	0.886 ***
	(0.072)	(0.070)	(0.193)	(0.193)	(0.070)	(0.069)	(0.073)	(0.073)
cesd	0.263	0.364 *	0.170	0.326	0.981 ***	0.935 ***	1.597 ***	1.435 ***
	(0.157)	(0.144)	(0.467)	(0.426)	(0.153)	(0.146)	(0.139)	(0.135)
cogtot	-0.016 *	-0.016 *	-0.016	0.004	0.007	0.004	-0.023 **	-0.022 **
	(0.008)	(0.008)	(0.024)	(0.023)	(0.009)	(0.008)	(0.008)	(0.007)
cohort5_48	0.074	0.101	-0.483	-0.484	-0.113	-0.043	-0.054	-0.048
	(0.082)	(0.080)	(0.297)	(0.281)	(0.090)	(0.088)	(0.084)	(0.083)
cohort6_54_59	0.194 *	0.043	-0.149	-0.525	-0.003	-0.031	-0.061	-0.027
	(0.090)	(0.090)	(0.302)	(0.318)	(0.100)	(0.098)	(0.094)	(0.091)
N	7775	7775	7775	7775	7775	7775	7775	7775
Mahalanobis distance	7.528	7.528	5.393	5.393	8.903	8.903	9.954	9.954
p-value	0.873	0.873	0.966	0.966	0.780	0.780	0.698	0.698

***** $p < 0.001$; ** $p < 0.01$; * $p < 0.05$. Standard errors are in parentheses. Low p-value (e.g., $<$ 0.01) reported below the Mahalanobis distance provides strong evidence against the null hypothesis that the parameter estimates from the real data and synthetic data are equal.**

Table 9: Contrasting parameter estimates of multinomial logistic regression model: real data versus synthetic data generated by CTGAN model

	2-Cardiovas		3-Cancer		4-other		5-Comorbid	
data type	**original**	**synthetic**	**original**	**synthetic**	**original**	**synthetic**	**original**	**synthetic**
(Intercept)	-0.449 *	3.235 ***	-4.536 ***	-0.604	-1.090 ***	1.138 **	-0.280	2.021 ***
	(0.228)	(0.303)	(0.922)	(0.470)	(0.251)	(0.358)	(0.228)	(0.359)
White	-0.096	-0.665 ***	1.422	0.294	0.447 **	-0.231	0.354 **	-0.306 *
	(0.128)	(0.113)	(0.726)	(0.191)	(0.152)	(0.136)	(0.137)	(0.137)
Black	0.512 ***	0.230 *	1.224	-0.712 ***	0.239	0.412 **	0.590 ***	-0.316 *
	(0.141)	(0.116)	(0.764)	(0.203)	(0.173)	(0.136)	(0.150)	(0.144)
Female	-0.123	-0.207 *	0.988 ***	0.772 ***	0.590 ***	0.181	0.499 ***	0.184
	(0.066)	(0.085)	(0.222)	(0.160)	(0.072)	(0.106)	(0.067)	(0.108)
cSES2_R	-0.104	-0.652 ***	-0.599	-0.841 ***	-0.188	-1.037 ***	-0.257 **	-1.146 ***
	(0.093)	(0.128)	(0.325)	(0.218)	(0.098)	(0.172)	(0.099)	(0.215)
RTHLTHCH_R	0.060	-0.481 ***	0.235	0.221	-0.319 ***	-0.205	-0.414 ***	-0.766 ***
	(0.084)	(0.110)	(0.266)	(0.179)	(0.083)	(0.130)	(0.077)	(0.121)
College	-0.115	-0.402 ***	0.043	-0.573 ***	0.070	-0.438 ***	-0.152	-1.096 ***
	(0.080)	(0.109)	(0.241)	(0.163)	(0.084)	(0.133)	(0.085)	(0.171)
behav_smoke3	0.103	0.142 *	0.152	-0.397 ***	0.343 ***	0.023	0.317 ***	0.044
	(0.065)	(0.072)	(0.192)	(0.107)	(0.069)	(0.085)	(0.066)	(0.088)
cHOME_R	-0.012	0.072	-0.118	0.546 ***	-0.215 *	-0.309 *	-0.219 **	-0.524 ***
	(0.076)	(0.109)	(0.232)	(0.142)	(0.084)	(0.138)	(0.080)	(0.154)
bmiH	0.717 ***	0.181 *	-0.235	-0.190	0.147 *	-0.395 ***	0.941 ***	0.709 ***
	(0.072)	(0.071)	(0.193)	(0.104)	(0.070)	(0.084)	(0.073)	(0.089)
cesd	0.263	1.324 ***	0.170	0.854 *	0.981 ***	1.382 ***	1.597 ***	1.497 ***
	(0.157)	(0.305)	(0.467)	(0.429)	(0.153)	(0.341)	(0.139)	(0.342)
cogtot	-0.016 *	-0.043 ***	-0.016	-0.029 *	0.007	-0.009	-0.023 **	-0.046 ***
	(0.008)	(0.010)	(0.024)	(0.015)	(0.009)	(0.012)	(0.008)	(0.012)
cohort5_48	0.074	0.792 ***	-0.483	0.792 ***	-0.113	0.554 ***	-0.054	0.706 ***
	(0.082)	(0.104)	(0.297)	(0.147)	(0.090)	(0.122)	(0.084)	(0.121)
cohort6_54_59	0.194 *	0.373 ***	-0.149	0.382 **	-0.003	0.384 ***	-0.061	0.272 *
	(0.090)	(0.096)	(0.302)	(0.144)	(0.100)	(0.111)	(0.094)	(0.119)
N	7775	7775	7775	7775	7775	7775	7775	7775
Mahalanobis distance	781.452	781.452	323.772	323.772	200.363	200.363	149.634	149.634
p-value	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

***** $p < 0.001$; ** $p < 0.01$; * $p < 0.05$. Standard errors are in parentheses. Low p-value (e.g., $<$ 0.01) reported below the Mahalanobis distance provides strong evidence against the null hypothesis that the parameter estimates from the real data and synthetic data are equal.**

7 Conclusion

Micro-level data is essential for high-quality modeling in sectors like healthcare and finance, yet strict privacy mandates often prevent organizations from sharing this information publicly. As a solution, synthetic tabular data—comprising numerical, categorical, and ordinal variables—has emerged as a powerful alternative that mirrors original statistical distributions while protecting individual identities, enabling evidence-based analysis without compromising privacy or utility. A central challenge in this field is navigating the inherent trade-off between data utility and privacy preservation; researchers strive to identify Pareto superior models that provide higher levels of both utility and privacy protection. While the literature offers a variety of metrics and frameworks, modern Generative AI architectures—specifically Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), Large Language Models (LLMs), and Diffusion models—are increasingly recognized for their ability to outperform conventional synthetic data generation techniques.

An important question that remains largely unexplored is: how sensitive are evidence-based econometric model policies to the substitution of real data with synthetic data? What metrics should be used to compare synthetic datasets generated by various models? These are the main issues addressed in this paper.

This paper first explains the differential privacy framework of Dwork and colleagues (Dwork, 2008; Dwork and Roth, 2014) and the DP-SGD (Differential Privacy Stochastic Gradient Descent) algorithm of Abadi et al. (2016) that various generative models can incorporate in their neural network training algorithms to achieve a guaranteed level of differential privacy. The paper then explains the main mechanics of GANs, VAEs, and Diffusion models for synthetic tabular data. After briefly describing the mechanics of GANs and VAEs, the paper details the mechanics of the discrete time diffusion model, Denoising Diffusion Probabilistic Models (DDPMs) (Sohl-Dickstein et al., 2015; Ho et al., 2020).

The paper then describes the elegant continuous time stochastic differential equation (SDE) framework that unifies discrete time diffusion models. In this framework, vector fields and diffusion coefficients are used for forward noising processes of the data points, and the Fokker-Planck equation (also known as the Kolmogorov forward equation) is employed to simplify the backward denoising process to approximate the data generating probability distribution and the algorithms for training and sampling.

The paper proposes the use of the Mahalanobis distance D^2 statistic as a measure of policy sensitivity to the substitution of real data with synthetic data for the statistical parameter estimates of econometric policy models.

The paper considers 13 tabular generative models—3 GAN-based, 2 VAE-based, 5 Diffusion-based, and 3 other types—to train and generate samples from each model, compute various utility and privacy metrics, and apply the proposed Mahalanobis distance metric for policy sensitivity to rank models.[1]

The paper finds that the best five models, ranked in decreasing order, are as follows:

- By the commonly used utility metric pMSE (propensity mean squared error), the best five models are ARF, CDTD, TabDDPM, SMOTE, and Tabsyn.
- By the privacy metric grMDCR (median Gower distance to closest record), the best five models are CTGAN, TVAE, SMOTE, CoDi, and ARF.
- By the weighted privacy metric in Lautrup et al. (2024) (that combines many individual metrics), the best five models are CTGAN, TVAE, CoDi, ARF, and TTVAE.

According to the proposed Mahalanobis D^2 metric, the models with statistically no policy sensitivity are ARF, CDTD, and TabDIFF for three econometric models of early childhood factors, and ARF, CDTD, and TabDDPM for the econometric model of midlife chronic disease incidence. The models that produce no statistically significant policy sensitivity in all econometric models considered in the paper are **ARF** and **CDTD**.

The synthetic data generator **ARF** stands out as the best compromise from the viewpoint of utility, privacy, and policy sensitivity metrics.

[1] **GAN Based models:** CTGAN (Xu et al., 2019), CTABGAN (Z. Zhao et al., 2021), CopulaGAN (Patki et al., 2021); **VAE based models:** TVAE (Xu et al., 2019), TTVAE (A. X. Wang and Nguyen, 2025); **Diffusion models:** TabDDPM (Kotelnikov et al., 2023), CoDi (Lee et al., 2023), Tabsyn (Zhang et al., 2024), TabDiff (J. Shi et al., 2024), CDTD (Mueller et al., 2023); **Other type of models:** SMOTE (Chawla et al., 2002), ARF (Watson et al., 2022), TabularARGN (Tiwald et al., 2025).

References

[1] Abadi, M. et al. “Deep Learning with Differential Privacy”, *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. CCS’16. ACM, Oct. 2016, 308–318. DOI: 10.1145/2976749.2978318 (cit. on pp. 5, 7, 43).

[2] Albergo, M. S. and Vanden-Eijnden, E. Building Normalizing Flows with Stochastic Interpolants, (Sept. 2022). DOI: 10.48550/ARXIV.2209.15571 (cit. on p. 14).

[3] Altae-Tran, H. et al. Low Data Drug Discovery with One-Shot Learning, *ACS Central Science*, **3**, no. 4 (2017), 283–293. DOI: 10.1021/acscentsci.6b00367 (cit. on p. 3).

[4] Barocas, S. and Selbst, A. D. Big data’s disparate impact, *Calif. L. Rev.*, **104** (2016), 671. DOI: 10.2139/ssrn.2477899 (cit. on p. 4).

[5] Capasso, M. Synthetic data as meaningful data. On Responsibility in data ecosystems, *Big Data & Society*, **12**, no. 4 (2025), 20539517251386053. DOI: 10.1177/20539517251386053 (cit. on p. 4).

[6] Chan, S. H. Tutorial on Diffusion Models for Imaging and Vision, (2024). DOI: 10.48550/arXiv.2403.18103 (cit. on p. 11).

[7] Chawla, N. V. et al. SMOTE: Synthetic Minority Over-sampling Technique, *Journal of Artificial Intelligence Research*, **16** (June 2002), 321–357. DOI: 10.1613/jair.953 (cit. on pp. 3, 21, 44).

[8] Dwork, C. “Differential Privacy: A Survey of Results”, *Theory and Applications of Models of Computation*. Ed. by M. Agrawal et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, 1–19 (cit. on pp. 5, 6, 43).

[9] Dwork, C. and Roth, A. The Algorithmic Foundations of Differential Privacy, *Foundations and Trends in Theoretical Computer Science*, **9**, no. 3–4 (2014), 211–407. DOI: 10.1561/0400000042 (cit. on pp. 5, 43).

[10] Fang, M. L. et al. “DP-CTGAN: Differentially Private Medical Data Generation Using CTGANs”, *Artificial Intelligence in Medicine*. Ed. by M. Michalowski et al. Cham: Springer International Publishing, 2022, 178–188. DOI: 10.1007/978-3-031-09342-5_17 (cit. on p. 8).

[11] Fisher, G. G. and Ryan, L. H. Overview of the Health and Retirement Study and Introduction to the Special Issue, *Work, Aging and Retirement*, **4**, no. 1 (Dec. 2017). Ed. by M. Wang, 1–9. DOI: 10.1093/workar/wax032 (cit. on p. 27).

[12] Fonseca, J. and Bacao, F. Tabular and latent space synthetic data generation: a literature review, *Journal of Big Data*, **10**, no. 1 (July 2023). DOI: 10.1186/s40537-023-00792-7 (cit. on p. 4).

[13] Giuffrè, M. and Shung, D. L. Harnessing the power of synthetic data in healthcare: innovation, application, and privacy, *npj Digital Medicine*, **6**, no. 1 (Oct. 2023). DOI: 10.1038/s41746-023-00927-3 (cit. on p. 4).

[14] Goodfellow, I. J. et al. Generative Adversarial Networks, (2014). DOI: 10.48550/ARXIV.1406.2661 (cit. on pp. 4, 8).

[15] He, H. and Garcia, E. A. Learning from Imbalanced Data, *IEEE Transactions on Knowledge and Data Engineering*, **21**, no. 9 (Sept. 2009), 1263–1284. DOI: 10.1109/TKDE.2008.239 (cit. on p. 3).

[16] Heckman, J. J. Schools, Skills and Synapses, *Economic Inquiry*, **46**, no. 3 (July 2008), 289–324. DOI: 10.1111/j.1465-7295.2008.00163.x (cit. on p. 35).

[17] Heckman, J. J. and Raut, L. K. Intergenerational long-term effects of preschool-structural estimates from a discrete dynamic programming model, *Journal of Econometrics*, **191**, no. 1 (2016), 164–175. DOI: 10.1016/j.jeconom.2015.10.001 (cit. on p. 35).

[18] Ho, J. et al. Denoising Diffusion Probabilistic Models, (June 2020). DOI: 10.48550/ARXIV.2006.11239 (cit. on pp. 4, 11, 43).

[19] Holderrieth, P. and Erives, E. An Introduction to Flow Matching and Diffusion Models, (June 2025). DOI: 10.48550/ARXIV.2506.02070 (cit. on p. 15).

[20] Johnson, R. A. and Wichern, D. W. Applied Multivariate Statistical Analysis. 6th ed. Harlow, Essex, England: Pearson Education Limited, 2013 (cit. on p. 26).

[21] Jordon, J. et al. "PATE-GAN: Generating Synthetic Data with Differential Privacy Guarantees", *International Conference on Learning Representations*. 2018 (cit. on p. 8).

[22] Juster, F. T. and Suzman, R. An Overview of the Health and Retirement Study, *The Journal of Human Resources*, **30** (1995), S7. DOI: 10.2307/146277 (cit. on p. 27).

[23] Kingma, D. P. and Welling, M. Auto-Encoding Variational Bayes, (2013). DOI: https://doi.org/10.48550/arXiv.1312.6114 (cit. on pp. 4, 9).

[24] Kotelnikov, A. et al. "TabDDPM: Modelling Tabular Data with Diffusion Models", arXiv, Sept. 2023. DOI: 10.48550/arXiv.2209.15421 (cit. on pp. 4, 19, 20, 44).

[25] Koul, A. et al. Synthetic data, synthetic trust: navigating data challenges in the digital revolution, *The Lancet Digital Health* (Dec. 2025), 100924. DOI: 10.1016/j.landig.2025.100924 (cit. on p. 4).

[26] Lautrup, A. D. et al. Syntheval: a framework for detailed utility and privacy evaluation of tabular synthetic data, *Data Mining and Knowledge Discovery*, **39**, no. 1 (2024). DOI: 10.1007/s10618-024-01081-4 (cit. on pp. 29, 31, 44).

[27] Lee, C. et al. "CoDi: Co-evolving Contrastive Diffusion Models for Mixed-type Tabular Synthesis", *International Conference on Machine Learning*. arXiv, 2023. DOI: 10.48550/arXiv.2304.12654 (cit. on pp. 20, 44).

[28] Li, Z. et al. *Diffusion Models for Tabular Data: Challenges, Current Progress, and Future Directions*. Feb. 2025. DOI: 10.48550/ARXIV.2502.17119 (cit. on pp. 4, 20).

[29] Lipman, Y. et al. Flow Matching for Generative Modeling, (2023). DOI: 10.48550/ARXIV.2210.02747 (cit. on pp. 14, 20).

[30] Luo, C. *Understanding Diffusion Models: A Unified Perspective*. 2022. DOI: 10.48550/ARXIV.2208.11970 (cit. on p. 11).

[31] Matarazzo, A. and Torlone, R. A Survey on Large Language Models with some Insights on their Capabilities and Limitations, (Jan. 2025). DOI: 10.48550/ARXIV.2501.04040 (cit. on p. 4).

[32] Mattei, P.-A. and Frellsen, J. MIWAE: Deep Generative Modelling and Imputation of Incomplete Data, (Dec. 2018). DOI: 10.48550/ARXIV.1812.02633 (cit. on p. 3).

[33] Mohammed, S. et al. The effects of data quality on machine learning performance on tabular data, *Information Systems*, **132** (2025), 102549. DOI: https://doi.org/10.1016/j.is.2025.102549 (cit. on p. 4).

[34] Mueller, M. et al. Continuous Diffusion for Mixed-Type Tabular Data, (2023). DOI: 10.48550/ARXIV.2312.10431 (cit. on pp. 20, 44).

[35] Narayanan, A. and Shmatikov, V. “Robust De-anonymization of Large Sparse Datasets”, *2008 IEEE Symposium on Security and Privacy (sp 2008)*. 2008, 111–125. DOI: 10.1109/SP.2008.33 (cit. on p. 21).

[36] Patki, N. et al. CopulaGAN: Learning Copula Models with Generative Adversarial Networks, *arXiv preprint arXiv:2101.00598* (2021). DOI: 10.48550/arXiv.2101.00598 (cit. on pp. 20, 44).

[37] Perez-Cruz, F. “Kullback-Leibler divergence estimation of continuous distributions”, *2008 IEEE International Symposium on Information Theory*. IEEE, July 2008, 1666–1670. DOI: 10.1109/isit.2008.4595271 (cit. on p. 22).

[38] Raut, L. K. “Determinants and Predictions of Risks of Diseases in Mid Ages: Logistic Regression Versus Deep Neural Network Models”, *Practical Economic Analysis and Computation: A Festschrift in Honor of Professor Kirit Parikh*. Ed. by P. P. Ghosh et al. Singapore: Springer Nature Singapore, 2024, 263–284. DOI: 10.1007/978-981-97-6753-3_12 (cit. on pp. 27, 38).

[39] Raut, L. K. Early childhood factors and health pathways to disability and death in mid-ages — a multi-state time-to-event model, *Working Paper* (2024) (cit. on p. 35).

[40] Raut, L. K. Long-term Effects of Preschool on School Performance, Earnings and Social Mobility, *Studies in Microeconomics*, **6**, no. 1-2 (June 2018), 24–49. DOI: 10.1177/2321022218802023 (cit. on p. 35).

[41] Rencher, A. C. and Christensen, W. F. Methods of Multivariate Analysis. 3rd ed. John Wiley & Sons, 2012. DOI: 10.1002/9781118391684 (cit. on p. 26).

[42] Sajjadi, M. S. M. et al. “Assessing Generative Models via Precision and Recall”, *Advances in Neural Information Processing Systems*. Ed. by S. Bengio et al. Vol. 31. Curran Associates, Inc., 2018 (cit. on p. 5).

[43] Shi, J. et al. “TabDiff: a Multi-Modal Diffusion Model for Tabular Data Generation”, *Conference on Neural Information Processing Systems*. arXiv, 2024. DOI: 10.48550/arXiv.2410.20626 (cit. on pp. 20, 44).

[44] Shi, R. et al. A Comprehensive Survey of Synthetic Tabular Data Generation, (Apr. 2025). DOI: 10.48550/ARXIV.2504.16506 (cit. on p. 4).

[45] Sohl-Dickstein, J. et al. “Deep Unsupervised Learning using Nonequilibrium Thermodynamics”, *Proceedings of the 32nd International Conference on Machine Learning*. Ed. by F. Bach and D. Blei. Vol. 37. Proceedings of Machine Learning Research. Lille, France: PMLR, July 2015, 2256–2265. DOI: 10.48550/arXiv.1503.03585 (cit. on pp. 4, 11, 43).

[46] Song, Y. et al. “Score-Based Generative Modeling through Stochastic Differential Equations”, *International Conference on Learning Representations (ICLR)*. 2021. DOI: 10.48550/arXiv.2011.13456 (cit. on p. 14).

[47] Sonnega, A. et al. Cohort Profile: the Health and Retirement Study (HRS), *International Journal of Epidemiology*, **43**, no. 2 (Mar. 2014), 576–585. DOI: 10.1093/ije/dyu067 (cit. on p. 27).

[48] Steffick, D. E. Documentation of affective functioning measures in the Health and Retirement Study, *Ann Arbor, MI: University of Michigan* (2000) (cit. on p. 27).

[49] Sweeney, L. k-anonymity: A model for protecting privacy, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, **10**, no. 5 (2002), 557–570. DOI: 10.1142/S0218488502001648 (cit. on p. 21).

[50] Tiwald, P. et al. *TabularARGN: A Flexible and Efficient Auto-Regressive Framework for Generating High-Fidelity Synthetic Data*. ML synthetic data,TabularARGN. 2025. DOI: 10.48550/ARXIV.2501.12012 (cit. on pp. 21, 44).

[51] Truda, G. *Generating tabular datasets under differential privacy*. Survey of GenAI, TableDiffusion. 2023. DOI: 10.48550/ARXIV.2308.14784 (cit. on p. 4).

[52] Vallevik, V. B. et al. Can I trust my fake data – A comprehensive quality assessment framework for synthetic tabular data in healthcare, *International Journal of Medical Informatics*, **185** (2024), 105413. DOI: https://doi.org/10.1016/j.ijmedinf.2024.105413 (cit. on p. 4).

[53] Wang, A. X. and Nguyen, B. P. TTVAE: Transformer-based generative modeling for tabular data generation, *Artificial Intelligence*, **340** (2025), 104292. DOI: https://doi.org/10.1016/j.artint.2025.104292 (cit. on pp. 10, 20, 44).

[54] Wang, Q. et al. Divergence Estimation for Multidimensional Densities Via k-Nearest-Neighbor Distances, *IEEE Transactions on Information Theory*, **55**, no. 5 (May 2009), 2392–2405. DOI: 10.1109/TIT.2009.2016060 (cit. on p. 22).

[55] Watson, D. S. et al. Adversarial random forests for density estimation and generative modeling, *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS 2023)* (May 2022). DOI: 10.48550/ARXIV.2205.09435 (cit. on pp. 21, 44).

[56] Weggenmann, B. et al. "DP-VAE: Human-Readable Text Anonymization for Online Reviews with Differentially Private Variational Autoencoders", *Proceedings of the ACM Web Conference 2022*. WWW '22. Virtual Event, Lyon, France: Association for Computing Machinery, 2022, 721–731. DOI: 10.1145/3485447.3512232 (cit. on p. 10).

[57] Xu, L. et al. "Modeling Tabular Data Using Conditional GAN", *Conference on Neural Information Processing Systems*. 2019 (cit. on pp. 8, 20, 44).

[58] Zhang, H. et al. "Mixed-Type Tabular Data Synthesis with Score-based Diffusion in Latent Space", *International Conference on Learning Representations*. 2024. DOI: 10.48550/ARXIV.2310.09656 (cit. on pp. 4, 20, 44).

[59] Zhao, W. X. et al. A Survey of Large Language Models, *arXiv preprint arXiv:2303.18223* (2023). DOI: 10.48550/arXiv.2303.18223 (cit. on p. 4).

[60] Zhao, Z. et al. "CTAB-GAN: Effective Table Data Synthesizing", *Proceedings of The 13th Asian Conference on Machine Learning*. Ed. by V. N. Balasubramanian and I. Tsang. Vol. 157. Proceedings of Machine Learning Research. PMLR, 17–19 Nov 2021, 97–112 (cit. on pp. 20, 44).