UNIVERSITÀ DEGLI STUDI DI TRIESTE

UNIVERSITÀ DEGLI STUDI DI TRIESTE

XXXVIII CICLO DEL DOTTORATO DI RICERCA IN

Applied Data Science and Artificial Intelligence

Finanziato dall'Unione europea - NextGenerationEU
Funded by the European Union - NextGenerationEU

Co-finanziatore della borsa: Aindo SpA

Synthetic Tabular Data at the Intersection of AI and Privacy: A Formal Evaluation of Disclosure Risks and Protection Mechanisms

Settore scientifico-disciplinare: **INF/01**

DOTTORANDO / A
Milton Nicolás Plasencia Palacios

COORDINATORE
PROF. Francesco Pauli

SUPERVISORi DI TESI
PROF. Luca Bortolussi
Dott. Sebastiano Saccani

ANNO ACCADEMICO 2024/2025

UNIVERSITÀ DEGLI STUDI DI TRIESTE

APPLIED DATA SCIENCE & ARTIFICIAL INTELLIGENCE

Università degli Studi di Trieste

Ph.D. in Applied Data Science & Artificial Intelligence

XXXVIII cycle

Synthetic Tabular Data at the Intersection of AI and Privacy: A Formal Evaluation of Disclosure Risks and Protection Mechanisms

Candidate
Milton Nicolás Plasencia Palacios

Supervisors
Prof. Luca Bortolussi
Dott. Sebastiano Saccani

Abstract

The rapid advancement of generative artificial intelligence has positioned synthetic tabular data as a promising solution for privacy-preserving data sharing. By generating artificial records that mirror the statistical properties of sensitive datasets, organizations aim to navigate the restrictive barriers of data protection regulations such as the GDPR. However, the tension between data utility and privacy remains a critical challenge. Modern deep generative models are prone to memorizing training instances, potentially leaking sensitive information through "singling out," "linkability," or "inference" attacks—the three pillars of data anonymity defined by the Article 29 Working Party (WP29). This thesis investigates this intersection of AI and privacy, providing a formal evaluation of disclosure risks and proposing novel mechanisms for robust data protection.

The first major contribution of this research is the development of a systematic taxonomy and a rigorous evaluation framework for privacy metrics. Current assessment methods often lack standardization, making it difficult to compare the safety of different generative models. We address this by introducing an attack-based metrics framework that utilizes Contrastive Learning to identify vulnerable "outlier" records. By framing privacy as a membership and attribute inference problem, we demonstrate how contrastive loss can more efficiently detect records at high risk of disclosure compared to traditional distance-based heuristics. Furthermore, we establish a validation protocol using controlled "Risk Models"—such as Overfitting and Differential Privacy models—to empirically test the sensitivity and reliability of these metrics under varying levels of vulnerability.

The second core contribution is the introduction of a Hybrid Data Synthesis Pipeline. Recognizing that neither traditional anonymization nor pure deep learning models satisfy the dual requirement of high utility and formal safety, we propose a layered architecture. This pipeline first applies formal statistical disclosure control, specifically k-anonymity, to create a structurally sanitized data backbone. We then utilize state-of-the-art generative models, including CTGAN and REaLTabFormer, to learn from this anonymized distribution and restore the complex statistical correlations lost during the initial sanitization phase. This approach ensures that the generative model is fundamentally restricted from memorizing unique, sensitive records, thereby providing a “layered privacy assurance" guarantee.

Experimental results across diverse datasets demonstrate that while the hybrid approach successfully mitigates specific disclosure risks, it also highlights the inherent trade-offs between localized fidelity and global privacy. The findings suggest that while attack-

based metrics offer superior granularity in detecting leaks, the choice of data transition methods within the hybrid pipeline is critical to maintaining the utility-privacy frontier. Ultimately, this thesis provides a comprehensive roadmap for the deployment of synthetic data, offering both a standardized language for risk assessment and a pragmatic engineering solution for secure data synthesis in highly regulated environments.

Acknowledgment

I would like to express my sincere gratitude to my co-supervisors at Aindo, Gabriele Sgroi and Alexander Boudewijn, for their invaluable guidance and support throughout these three years; their sharp insights and critical feedback were instrumental in shaping my research.

Contents

Chapter 1

Introduction

In the contemporary era of the digital revolution, the accumulation of high-dimensional tabular data has reached an unprecedented scale, fundamentally transforming fields as diverse as precision medicine, algorithmic finance, and social science research. This data serves as the lifeblood of evidence-based policymaking and modern artificial intelligence, offering the potential to uncover hidden patterns that can solve some of society's most complex problems. From predicting disease outbreaks to optimizing global supply chains, the ability to analyze granular information about individuals and systems has become a cornerstone of technological progress. However, the vast majority of this information is inherently sensitive, containing personal details that are protected by strict ethical guidelines and rigorous legal frameworks. The introduction of the General Data Protection Regulation in Europe and similar mandates globally has created a significant hurdle for the scientific community. While the demand for data access has never been higher, the legal and technical risks associated with data sharing have never been more acute, leading to a state where much of the world's most valuable data remains siloed and inaccessible.

This tension has birthed a fundamental privacy-utility paradox within the field of data science. Historically, researchers have relied on traditional statistical disclosure control methods to sanitize datasets before they are released to the public or shared with third parties. Techniques such as data masking, generalization, and local perturbation were designed to remove direct identifiers and obscure sensitive values. Unfortunately, these classical methods often fail to meet the rigorous demands of modern machine learning. In the quest to protect individual privacy, traditional anonymization frequently destroys the subtle, non-linear correlations and complex multivariate distributions that deep learning models require to function accurately. When a dataset is rendered safe enough to comply with legal standards, it is often no longer useful enough for high-fidelity analysis. Conversely, when the statistical utility is preserved, the risk of re-identification through sophisticated linkage attacks remains unacceptably high, especially as computational power and the availability of external auxiliary data continue to grow.

To resolve this conflict, the field of Generative Artificial Intelligence has emerged as a transformative candidate for secure data dissemination. Synthetic Data Generation involves the training of deep generative models, such as Generative Adversarial Networks

and Variational Autoencoders, to learn the underlying probability distribution of a sensitive source dataset. Once these models have successfully captured the mathematical essence of the data, they can sample entirely new, artificial records. These records possess the same statistical properties and correlations as the original population but do not maintain a one-to-one correspondence with any real individual. In theory, this provides a mathematical twin of the data that is exempt from privacy restrictions because it does represent actual people, thereby allowing for the free flow of information without compromising individual confidentiality.

However, the transition from theoretical promise to practical deployment is fraught with hidden dangers. The very strength of deep learning, which is its ability to capture high-dimensional complexity with extreme precision, is also its primary weakness in the context of privacy. These models are designed to minimize reconstruction loss, a process that inherently encourages the memorization of training instances. This is particularly true for unique outliers who exist at the edges of the distribution. As a result, the synthetic output may inadvertently leak the very secrets it was intended to protect, acting as a compressed version of the original database rather than a generalized representation. This thesis explores this delicate intersection, moving beyond the initial promise of synthetic data to provide a rigorous, formal evaluation of whether these artificial datasets truly fulfill their mission of being both statistically representative and fundamentally private. It seeks to establish a new standard for how we measure risk and how we architect the next generation of privacy-preserving generative models.

The central obstacle in the current synthetic data landscape is the profound difficulty of accurately measuring privacy leakage within high-dimensional datasets. In the contemporary research environment, a significant disconnect exists between empirical privacy metrics and the legal definitions of anonymity provided by regulatory bodies. While the Article 29 Working Party defines the three pillars of privacy risk as Singling Out, Linkability, and Inference, these concepts do not always translate clearly into mathematical formulas. Most current evaluation methods rely on fragmented, distance-based heuristics that often fail to capture the sophisticated, non-linear relationships that deep learning models exploit. This results in an evaluation crisis where a generative model may appear safe under one set of metrics while remaining dangerously vulnerable to another, ultimately creating a false sense of security that can lead to catastrophic re-identification in real-world applications.

A secondary and equally pressing challenge involves the limitations of existing defense mechanisms, particularly the trade-offs inherent in formal privacy frameworks like Differential Privacy. Although Differential Privacy offers a mathematically rigorous guarantee against individual disclosure, its application to tabular data is notoriously problematic. Tabular datasets are characterized by rigid structural constraints, functional dependencies, and highly skewed distributions that are easily destroyed by the noise levels required to achieve a meaningful privacy budget. This often results in a scenario where the formally private data becomes analytically useless for downstream machine learning tasks, as the statistical signal is effectively drowned out by the protective noise. This binary tension forces organizations to choose between high-utility data that lacks

formal guarantees or safe data that lacks any practical value for researchers.

Furthermore, even when defensive layers are applied, they can introduce paradoxical vulnerabilities. Traditional anonymization techniques such as k-anonymity provide a structural backbone of safety but are often criticized for their inability to preserve the high-dimensional correlations necessary for complex data science. When researchers attempt to restore this lost utility through transition methods or conditional sampling, they risk re-injecting overly specific information that was originally suppressed. This transition paradox highlights the necessity of a more sophisticated, hybrid approach that can navigate the delicate frontier between privacy and utility. The fundamental problem addressed in this thesis is the lack of a cohesive framework that can both provide standardized, attack-based risk assessments and a generation architecture that offers structural protection without sacrificing the essential statistical integrity of the data.
To address the challenges outlined above, this thesis is guided by the following core research questions, which align with the methodological progression of the chapters:

- **RQ1**: How can the gap between high-level legal definitions of anonymity (Singling Out, Linkability, Inference) and empirical technical evaluations be bridged? We investigate whether current distance-based metrics are sufficient to detect complex privacy leaks and propose a novel taxonomy and Contrastive Learning-based framework to capture non-linear vulnerabilities.
- **RQ2**: How can the reliability and sensitivity of privacy metrics be rigorously verified in the absence of a ground truth? We hypothesize that by injecting controlled vulnerabilities—through Risk Models such as overfitting, leakage, and noise—we can empirically benchmark the "detection power" of privacy metrics, distinguishing effective auditing tools from those that provide a false sense of security.
- **RQ3**: Can a hybrid architecture combining Statistical Disclosure Control (SDC) with Deep Generative Models overcome the limitations of using either approach in isolation? We examine whether using k-anonymity as a structural backbone for generative models can provide a "privacy-by-design" guarantee that purely stochastic models lack, while still preserving higher utility than traditional anonymization.

1.1 Contribution

To provide a clear understanding of the original research presented in this work, the following section delineates the core advancements made to the fields of privacy-preserving machine learning and synthetic data evaluation. These contributions represent a shift from purely heuristic-based assessments toward a formalized, attack-driven methodology and a structural rethinking of how generative models can be safely architected for sensitive tabular environments.

The first major contribution of this thesis is a novel enhancement to the state-of-the-art Anonymeter framework through the integration of a Contrastive Learning-based solution. Recognizing that traditional distance-based metrics often struggle to capture the

non-linear and high-dimensional ways in which deep learning models leak information, this approach focuses on generating rich latent representations of tabular data. By mapping raw records into a semantic embedding space, the framework enables the detection of subtle and complex multi-attribute vulnerabilities that are typically invisible to standard evaluation tools. We demonstrate that executing density-based outlier detection, specifically Local Outlier Factor (LOF), within this learned latent space significantly improves the sensitivity of the Singling Out attack. This methodology provides a more rigorous and statistically sound measure of model memorization, allowing for the identification of specific records that have been effectively "hard-coded" into the generative model's parameters. This advancement moves the evaluation of synthetic data beyond simple proximity checks and into a dynamic simulation of informed adversarial behavior, ensuring that even the most nuanced privacy leaks are accounted for.

Building upon this evaluative framework, a second significant contribution is the creation of a systematic taxonomy of privacy metrics mapped directly to international regulatory standards. While previous literature has offered a fragmented view of privacy measurement, this work organizes these diverse metrics according to the "three pillars" of anonymity defined by the Article 29 Working Party: Singling out, Linkability, and Inference. This taxonomy serves as a bridge between high-level legal requirements and concrete mathematical implementation, providing a standardized language for auditors and data scientists alike. To validate the reliability of this taxonomy, we developed a suite of controlled "Risk Models"—including models intentionally subjected to varying degrees of overfitting and differential privacy noise. By testing our metrics against these controlled environments, we have empirically proven their sensitivity and reliability, establishing a rigorous protocol for certifying the safety of synthetic data generators before they are deployed in production.

The third and perhaps most significant constructive contribution is the design and development of the Hybrid Data Synthesis Pipeline. This novel architecture addresses the inherent "utility-privacy gap" found in current state-of-the-art models. Rather than relying solely on a generative model to learn privacy through noise—as is the case with standard Differentially Private GANs—the hybrid pipeline introduces a layered defense strategy. It utilizes traditional statistical disclosure control, specifically a robust implementation of k-anonymity, to create a structurally sanitized backbone of the sensitive data. This sanitized data then serves as the training input for advanced generative models like CTGAN and REaLTabFormer. This strategy ensures a "privacy-by-design" guarantee: the generative model is mathematically prevented from ever seeing or memorizing unique individual records, yet it remains free to learn and replicate the complex, multivariate correlations necessary for high-utility data analysis.

Finally, this research contributes a detailed investigation into the "transition paradox" and the intricacies of high-fidelity sampling in synthetic data. Through extensive experimentation, we have identified how certain sampling strategies—intended to restore utility lost during anonymization—can inadvertently re-introduce localized privacy risks. By analyzing the behavior of Single, Uniform, and Conditional transitions, this work provides the first comprehensive look at how the mechanics of value-sampling affect

the final privacy-utility frontier. These findings offer actionable guidelines for practitioners, demonstrating that the safety of a synthetic dataset is determined not just by the training algorithm, but by the entire pipeline of data transformation. Collectively, these contributions provide both the analytical tools to measure risk and the engineering frameworks to mitigate it, paving the way for the ethical and secure use of artificial intelligence in sensitive data domains.

The development of these insights was significantly enriched by a collaborative research period at KU Leuven, conducted under the supervision of Professor Vincent Naessens, Dott. Michiel Willocx, and Dott. Kevin De Boeck. It was during this visiting stay that the preliminary findings regarding sampling-induced risks were first derived, benefiting from the specialized expertise in secure data processing at the host institution.

The results presented herein are the product of this international collaboration and are based on a forthcoming publication co-authored with Niels Van Bossche. This work reflects the substantial and equal scientific contributions made by both PhD students, representing a unified effort to address the complexities of high-fidelity data synthesis.

In addition to the work presented in this thesis, I am currently collaborating on an ongoing project supervised by Professor Alejandro Rodriguez Garcia. This research focuses on the development of "Spectral Density Peaks" (SDP), a project primarily led by PhD student Andrea Mecchina, with significant support from myself and fellow PhD student Francesco Tomba. Spectral Density Peaks (SDP) is a novel unsupervised learning framework that integrates the rigorous graph-theoretical foundations of Spectral Clustering with the adaptive flexibility of density-based methods. By leveraging an intrinsic probabilistic formulation that accounts for density estimates and their relative uncertainties, the algorithm effectively identifies clusters of arbitrary shapes without being restricted to the convex geometries that limit k-means. This unified approach addresses common drawbacks such as the need for pre-specifying cluster counts and sensitivity to noise, providing a more robust tool for analyzing complex datasets. Extensively validated against state-of-the-art competitors, SDP demonstrates superior performance across both simulated and real-world scenarios while offering an accessible Python implementation for practical application.

1.2 Thesis Structure

The thesis is structured as follows:

- Chapter 2: Preliminaries establishes the necessary theoretical and technical foundations, reviewing Generative AI, Deep Learning, traditional data anonymization models (k-anonymity, l-diversity, t-closeness), and the concept of Differential Privacy (DP) as applied to synthetic data generation.
- Chapter 3: Attack-based Metrics Framework details our proposed enhancement to the Anonymeter privacy auditing framework, focusing on the Contrastive Learning-based method for improving the detection sensitivity of Singling Out, Linkability, and Inference attacks.

- Chapter 4: Taxonomy of Privacy Metrics presents our comprehensive classification of privacy quantification methods and introduces the novel Risk Models employed for systematic empirical evaluation of metric efficacy.
- Chapter 5: Hybrid pipeline for Synthetic Data Generation introduces the novel three-stage methodology designed to leverage the complementary strengths of anonymization and SDG, presenting preliminary results and architectural flexibility analysis.
- Chapter 6: Conclusions synthesizes the key findings from the empirical studies, outlines the achieved advances in privacy assurance, and proposes critical avenues for future research.

1.3 Research Papers

The contributions of this thesis are based on the following research papers:

- "Contrastive Learning-Based privacy metrics in Tabular Synthetic Datasets" [99], Milton Nicolás Plasencia Palacios, Sebastiano Saccani, Gabriele Sgroi, Alexander Boudewijn, Luca Bortolussi, arXiv preprint, 2025;
- "Empirical Evaluation of Structured Synthetic Data Privacy Metrics: Novel experimental framework " [98], Milton Nicolás Plasencia Palacios, Alexander Boudewijn, Sebastiano Saccani, Andrea Filippo Ferraris, Diana Sofronieva, Giuseppe D'Acquisto, Filiberto Brozzetti, Daniele Panfilo, Luca Bortolussi, arXiv preprint, 2025;
- "From Generalization to Generation: Combining Anonymization and Synthesization in a Hybrid Privacy-Preserving Data Pipeline", Niels Van Bossche*, Milton Nicolás Plasencia Palacios*, Kevin De Boeck, Michiel Willocx, Vincent Naessens, Manuscript in progress.

Other contribution not included in this thesis:

- "Spectral Density Peaks Clustering via Perron Cluster Analysis", Andrea Mecchina, Francesco Tomba, Milton Nicolás Plasencia Palacios, Alejandro Rodriguez Garcia, Manuscript in progress.

1.4 Notation and Definitions

To ensure clarity and precision throughout this thesis, we adopt the following mathematical notation and definitions, inspired by the formalization in [101].

1.4.1 General Notation

We denote the set of real numbers by $\mathbb{R}$ and the set of natural numbers by $\mathbb{N}$. Probability density functions are denoted by $\mathbb{P}$.

*Equal contribution.

1.4.2 Datasets and Attributes

Let $\mathcal{D}$ denote a dataset (database) containing records sampled from a population P. We define the structure of the data as follows:

- **Attributes:** Let $\mathcal{A}(\mathcal{D})$ be the set of attributes (columns) in the dataset.
- **Records:** A record (or row) $d \in \mathcal{D}$ is defined as a tuple of values corresponding to the attributes in $\mathcal{A}(\mathcal{D})$.
- **Values:** We denote the value of a specific attribute $a \in \mathcal{A}(\mathcal{D})$ for a record d as $v(d, a)$.

Attributes are classified based on their domain:

- **Numerical Attributes:** An attribute a is numeric if its domain is continuous or ordered discrete (e.g., $v(d, a) \in \mathbb{R}$).
- **Categorical Attributes:** An attribute a is categorical if it takes values from a finite set of discrete categories.

1.4.3 Synthetic Data Generation

We define the process of synthetic data generation using the following notation:

- **Generative Model (G):** A stochastic function or algorithm capable of generating new data points.
- **Trained Generator ($G(\mathcal{D})$):** A generative model trained on the real dataset $\mathcal{D}$.
- **Synthetic Dataset ($\hat{\mathcal{D}}$):** A dataset generated by sampling from the trained model, denoted as $\hat{\mathcal{D}} \sim G(\mathcal{D})$. The records $\hat{d} \in \hat{\mathcal{D}}$ are referred to as synthetic records.

Chapter 2

Preliminaries

2.1 Introduction to Generative Artificial Intelligence

The pursuit of Artificial Intelligence (AI) is often traced back to the seminal workshop held at Dartmouth College in 1956, organized by Marvin Minsky and John McCarthy. The workshop's original ambition was to explore how machines might simulate aspects of human intelligence, such as reasoning, problem-solving, and language use.
In its nascent stages, the field was dominated by a symbolic approach, often referred to as "Good Old-Fashioned AI" (GOFAI). These early systems relied on explicit, hand-coded rules and logic to manipulate symbols. While effective in controlled environments, these rule-based systems proved brittle and incapable of handling the ambiguity and complexity of the real world. This limitation, coupled with the high cost of maintenance and a lack of sufficient computational power, led to a period of reduced funding and skepticism in the 1970s, widely known as the "AI Winter".
The landscape began to shift significantly in the 90s with the advent of the statistical revolution. Rather than relying on static, human-engineered rules, researchers turned toward data-driven approaches. This era marked the formalization of *Machine Learning* (ML), a paradigm where algorithms are designed to learn patterns from empirical data. Standard ML workflows formalized the practice of splitting data into training and test sets to rigorously evaluate a model's ability to generalize to unseen examples.
As computational power improved—specifically through the evolution of processing units in the 90s and the subsequent rise of GPUs, complex models such as Artificial Neural Networks (ANN) became computationally feasible. This resurgence laid the foundation for *Deep Learning*, a subfield that has since become the backbone of contemporary AI.
Within this modern landscape, a critical distinction must be drawn between two fundamental modeling approaches: *Discriminative* and *Generative* AI.

- **Discriminative AI** focuses on learning the boundary between classes. Formally, given an input X (e.g., an image) and a label Y (e.g., "cat" or "dog"), discriminative models attempt to learn the conditional probability distribution $P(Y|X)$. Their primary goal is mapping input features to a target label, making them highly effective for classification and regression tasks.

- **Generative AI**, the focus of this thesis, aims to solve a more complex problem:

modeling the underlying structure of the data itself. Instead of focusing solely on the boundary between classes, generative models estimate the joint probability distribution $\mathbb{P}(X, Y)$ or simply the distribution of the data $\mathbb{P}(X)$ in unsupervised settings. By learning the probability distribution from which the training data was sampled, these models acquire the ability to sample new, synthetic data points that are statistically similar to the original dataset.

While generative methods have existed for decades (for example, Naive Bayes or Gaussian Mixture Models), the field has recently undergone a paradigm shift. The integration of Deep Learning has enabled the creation of *Deep Generative Models*, such as Generative Adversarial Networks (GANs) and Transformers. These architectures are capable of capturing high-dimensional, complex correlations in unstructured data (images, text) and structured data (tabular records) that were previously intractable.
However, these advanced generative capabilities rely entirely on the optimization techniques and architectural components established by Deep Learning. Therefore, to understand how synthetic data is generated, we must first review the foundational concepts of neural architectures, optimization strategies, and learning mechanisms that enable these systems to function.

2.1.1 Deep Learning

Deep Learning (DL) [76] is a subfield of machine learning that focuses on training models composed of multiple hierarchical layers to automatically learn increasingly abstract and representative features from raw data. Unlike traditional approaches, where features often had to be hand-engineered, deep learning enables models to discover relevant representations directly from data, making it especially powerful for complex tasks such as image recognition, natural language processing, and speech understanding.
At the core of these models is the definition of a specific learning task, which dictates how the model interprets data. The most common objective is *classification*, where the network learns to assign input data to discrete categories, such as identifying whether an image contains a specific object. Other fundamental tasks include *regression*, aimed at predicting continuous numerical values, and *generative modeling*. By defining a *loss function* that quantifies the error between the model's prediction and the actual target, the network can iteratively adjust its internal parameters to improve performance on the task at hand.

The foundation model used to execute these tasks is the *Artificial Neural Network*. The conceptual foundation of artificial neurons was established in 1943 by McCulloch and Pitts [88].
Formally, given the input data $\mathbf{x} \in \mathbb{R}^n$, a neuron computes a non-linear transformation of it as follows:

$$y = f(\mathbf{W} \cdot \mathbf{x} + \mathbf{b}) \tag{2.1}$$

where $\mathbf{W}$ and $\mathbf{b}$ are called *weights* and *bias* and are the learnable parameters. As illustrated in Figure 2.1, the weights represent the strength of the connections from the input features, while the bias allows the activation function to be shifted, providing the model with the flexibility to fit data that does not pass through the origin.

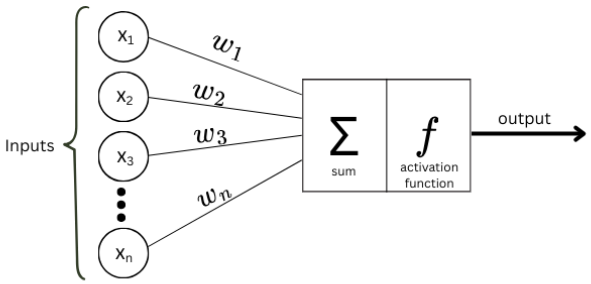


Figure 2.1: Schematic diagram illustrating the structure of a single artificial neuron, showing how weighted inputs are summed and processed through an activation function to generate an output.

In practice, the bias term is frequently incorporated into the weight vector by augmenting the input $\mathbf{x}$ with an additional constant feature (typically set to 1). This "absorbing bias" reparameterization simplifies the computation to a single dot product, $\mathbf{W}' \cdot \mathbf{x}'$, where $\mathbf{W}' = [\mathbf{W}, \mathbf{b}]$ and $\mathbf{x}' = [\mathbf{x}, \mathbf{1}]$. This notation is common in literature as it streamlines the mathematical exposition of gradient descent.
The function f is a non-linear function known as the *activation function*. Its role is to introduce non-linearity into the model, enabling the network to capture complex relationships that linear models cannot achieve. The choice of activation function is typically dictated by the specific role of the neuron within the architecture or the nature of the learning task. For instance, the *Rectified Linear Unit* (ReLU), defined as:

$$\text{ReLU}(x) = \max(0, x), \tag{2.2}$$

is the standard choice for hidden layers because it mitigates the vanishing gradient problem and accelerates convergence. In contrast, for the output layer of a binary classification task, a *Sigmoid* function is preferred because it squashes the output into a range of $[0, 1]$, which can be interpreted as a probability. Other functions like *Tanh* are often used when zero-centered data distributions are required in intermediate layers.

The first practical learning algorithm utilizing the neural structure was the *perceptron*, proposed by Frank Rosenblatt in 1958 [113]. The original perceptron was a single-layer architecture capable of learning linear decision boundaries. However, this limitation was later overcome by the *Multi-Layer Perceptron* (MLP), which consists of an ensemble of neurons arranged in layers.
In an MLP, the layers situated between the input and output are known as *hidden layers*. Each neuron in a given layer exchanges information with the neurons in the previous and subsequent layers, while no information is shared between neurons within the same layer. These hidden layers, coupled with non-linear activation functions, allow the MLP to act as a universal function approximator [61], enabling the model to represent highly complex, non-linear relationships.

During tasks like binary classification, the network processes inputs (e.g., using a sigmoid function) to output a class probability. The accuracy relies on parameters $\mathbf{W}$ and $\mathbf{b}$, which are optimized by minimizing a *loss function* that quantifies the prediction error. In the context of regression, a common choice is the *mean squared error* (MSE):

$$\mathcal{L}_{\text{regr}}(\mathbf{w}) = \frac{1}{N}\sum_{i=1}^{N}(y_i - f(x_i;\mathbf{w}))^2 \tag{2.3}$$

Conversely, for binary classification, the *binary cross-entropy loss* is widely used:

$$\mathcal{L}_{\text{class}} = \frac{1}{N}\sum_{i=1}^{N}\left[-y_i \log(f(x_i;\mathbf{w})) + (1-y_i)\log(1-f(x_i;\mathbf{w}))\right] \tag{2.4}$$

Once the loss is defined, the parameters are iteratively adjusted using the *backpropagation algorithm*. Popularized in 1986 [116], backpropagation computes the gradient of the loss function with respect to each parameter by applying the chain rule of calculus, allowing error signals to propagate backward from the output layer through the hidden layers. These gradients are used to update the weights via *gradient descent*:

$$\mathbf{w} \leftarrow \mathbf{w} - \eta \nabla_{\mathbf{w}} \mathcal{L} \tag{2.5}$$

where η is the learning rate. In practice, *stochastic gradient descent* (SGD) is often preferred, where gradients are estimated using small batches of data to improve computational efficiency and help the model escape local minima.
While deep architectures offer significant representative power [10], they are susceptible to *overfitting*. This occurs when a model learns the intrinsic noise of the training data rather than the underlying pattern, resulting in excellent training performance but poor generalization to unseen data. To mitigate this, various regularization techniques are employed: *dropout* randomly deactivates neurons during training to prevent co-adaptation; *early stopping* halts training when validation performance plateaus; and *normalization* layers stabilize activations to prevent parameters from growing excessively large.
Finally, to ensure that the model generalizes well across different subsets of the data and to obtain a robust estimate of its performance, *cross-validation* techniques are frequently employed to partition the dataset into multiple folds for alternating training and validation phases [125].

2.1.2 Architectures

The MLP has been designed to work with numerical structured data and has shown limitations in modeling unstructured data like images or text. To address these challenges, specialized types of NN have been developed to better adapt to specific domains.
Convolutional Neural Networks (CNN) [96] are a type of neural network that is specifically designed for images. Its name derives from the convolution operator applied in each layer of the network, which allows the model to learn the local correlations between the pixels in the image. This enables CNNs to automatically capture spatial hierarchies in images, making them highly effective for tasks like image classification, object detection, and segmentation.

Recurrent Neural Networks (RNN) [119] are designed to detect and model relations in sequential data, such as time series or natural language. They achieve this through the *recurrent units*, which allow the network to retain information from previous inputs in a sequence. Over time, RNNs architecture evolved to improve their memory and mitigate issues like vanishing gradients [75]. Prominent examples include Gated Recurrent Units (GRUs) and Long Short-Term Memory (LSTM) networks [141], which provide mechanisms to selectively retain or forget information over longer sequences.

Transformers [132] represent the current state of the art model regarding text modeling. Unlike RNNs, which process data sequentially, Transformers process the entire sequence in parallel. In this context, the input text is first decomposed into *tokens*—the basic units of information which can represent words, sub-words, or characters depending on the chosen vocabulary. Transformers' novelty consists in the introduction of the *attention mechanism*, which allows the model to evaluate the importance of each token in the context of the preceding words. However, because Transformers process tokens in parallel, they lack an inherent sense of order. To address this, *Positional Encodings* are added to the input embeddings to provide information about the relative or absolute position of each token in the sequence. In practice, text is first tokenized into numerical representations; then, the attention layer determines how much each word is related to the previous ones. The transformer computes the attention scores for each word in the text with respect to the previous ones multiple times (multi-head attention) to capture multiple contextual relationships simultaneously. Finally, the outputs are combined to form a more comprehensive understanding of the sequence. While originally proposed for machine translation, Transformers have been adapted to numerous domains, including Image Processing (Vision Transformers) [31] and protein sequence prediction [14].

2.1.3 Non-Neural and Ensemble Methods

For completeness, it is important to note that not all artificial intelligence models are based on neural networks, as discussed in the previous section. Some of the most widely used non-neural approaches include Probabilistic AI [47], like Bayesian Networks [57] that enhance interpretability by handling uncertainty through probability distributions and tree-based techniques like Extreme Gradient Boosting (XGBoost) [25].

XGBoost is particularly relevant for the subsequent sections. In particular, it is an ensemble method, meaning that it trains weak learners (decision trees in this case) sequentially, and combines their results to get a single strong classifier or regressor. The algorithm improves predictions using a gradient-based optimization approach. Specifically, after each tree is trained, the residuals, defined as the differences between the ground truth and the model's predictions, are computed. These residuals are then used to guide the training of the next tree in order to minimize a specified loss function. By aggregating the contributions of all trees, XGBoost achieves highly accurate predictions while maintaining robustness against overfitting.

In the context of this research, XGBoost is used as a robust judge for evaluating the *utility* and *quality* of synthetic data (discussed in Section 2.3).

2.2 Tabular Data

Despite the recent focus on unstructured data like text and images, tabular (structured) data remains fundamental in domains like finance and healthcare. In tabular datasets, information is organized in tables where each row d_i represents an individual and each column a_j represents an attribute (Figure 2.2).

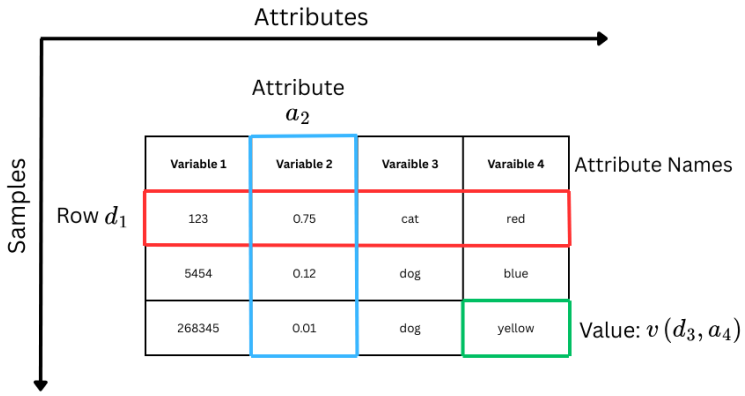


Figure 2.2: Example of a tabular dataset. The row denotes the individuals/samples, the columns are the attributes/features

In this format are present mixed data types that could be: integers, floats, strings, chars that can be categorized into numeric (continuous or discrete) and categorical (nominal or ordinal). This versatility makes tabular datasets suitable for modeling a wide range of real-world problems. However, the mixed data types also introduce challenges for machine learning models, as many algorithms are designed to operate only on numerical inputs. To address this, encoding techniques are employed to convert categorical or string-based variables into numerical representations. The most widely used encoders are:

- *Label Encoder*, which maps each value of the attribute into a number. While computationally efficient, this approach implicitly imposes an ordering on the categories, which may not be desirable.
- *One-Hot Encoder*, which represents each category as a binary vector, where a single entry is set to 1 and all others to 0. This method avoids imposing artificial order but can significantly increase dimensionality if the number of categories is large.

The choice of encoding method depends on the nature of the attribute and the requirements of the model. By applying appropriate encodings, machine learning models can effectively process tabular data for tasks such as classification or prediction of target attributes.
The most common task performed on tabular dataset is the classification or prediction of a specific attribute/column of the dataset. The application of ML models on this data type is of particular importance in many fields like healthcare, where a machine learning model can be able to predict disease risks; in finance to predict the future trends

in the stock markets or for fraud detection; and in industry for customer recommendation. In order to train ML models, tabular datasets are available on many repositories such as Kaggle. However, obtaining data for specific domains—especially in the medical field—can be difficult. These datasets often contain private or sensitive information, which cannot be freely shared due to privacy regulations and ethical concerns.

2.2.1 Introduction to Data Privacy and Anonymization

While the widespread availability of digital data drives research and innovation, sharing raw datasets raises significant privacy concerns. Safe data sharing must adhere to legal frameworks like the General Data Protection Regulation (GDPR) [38], which defines personal data as any direct or indirect information (e.g., physiological or economic traits) that can isolate and identify a natural person. This includes direct identifiers (e.g., names) and indirect factors (e.g., physiological or economic traits) that can isolate an individual. Anonymization, as clarified in Recital 26 [39], renders data non-personal by ensuring the data subject is no longer identifiable. This process is risk-based: identifiability is determined by considering all means 'reasonably likely to be used' by an attacker, accounting for cost, time, and available technology.

To systematically address these risks, it is essential to distinguish between different types of attributes based on their disclosure potential. *Identifiers* are attributes that directly reveal identity, such as a national ID number, and must be removed. In contrast, *Quasi-identifiers* (QIs) [104] are attributes like age, gender, and postal code that do not uniquely identify a person on their own but can do so when combined or linked with external data sources. Finally, *sensitive attributes* contain the confidential information intended for protection, such as medical diagnoses or income levels. It has been shown that even when direct identifiers are removed, individuals can often be re-identified by linking a combination of seemingly innocuous attributes with external data sources [126].

The failure to properly protect these attributes leads to specific types of privacy breaches, primarily categorized as *Identity Disclosure* and *Attribute Disclosure*. Identity disclosure occurs when an intruder can associate a specific record with a unique individual, often through linkage attacks. Attribute disclosure is more subtle; it occurs when an intruder learns new, confidential information about a data subject even if they cannot perfectly isolate that individual's specific record. For example, if an attacker knows a target is part of a group where every member has the same sensitive condition, the attribute is disclosed regardless of whether the specific record is identified.

To evaluate the success of anonymization, the Article 29 Working Party (WP29) [5] defines three technical risks that must be mitigated: *Singling Out*, which is the ability to isolate a record; *Linkability*, the ability to connect records across different databases; and *Inference*, the ability to deduce attribute values with high probability. These criteria directly inform the evolution of privacy-preserving models. The foundational model of k-anonymity was designed to prevent Singling Out and identity disclosure by ensuring each individual is indistinguishable from at least $k-1$ others. However, the subsequent development of l-diversity and t-closeness was necessary to address the specific vulnerability of attribute disclosure, ensuring that sensitive values within a group are sufficiently diverse or follow a safe distribution. These models are supported by practical techniques such as generalization, suppression, and sampling, which have been incorporated into

robust anonymization tools like ARX [106].
At its core, the field seeks to balance the conflicting goals of protecting privacy and retaining analytical value. While traditional anonymization filters information through generalization and suppression, Synthetic Data Generation (SGD) offers an alternative by creating entirely new records that mimic the statistical properties of the original data without being linked to real individuals. In both cases, the goal remains to meet the legal standard of Recital 26 by ensuring that re-identification is not "reasonably likely" given current and future technological developments.

k-Anonymity

According to the Article 29 Data Protection Working Party (WP29), anonymization is not a single outcome but a process involving different techniques to mitigate the risk of re-identification. These techniques are broadly categorized into three families: *pseudonymization*, *randomization*, and *generalization* [5]. While pseudonymization focuses on replacing direct identifiers (e.g., IDs or names) with artificial aliases, randomization and generalization aim to protect the dataset against the linking of individuals through quasi-identifiers.
Quasi-identifiers, as already explained previously, can lead to re-identification [126] if combined or linked with external datasets (e.g., a public voter list). To protect against linkage attacks, Samarati and Sweeney [117, 126] introduced k-anonymity, a model utilizing generalization and suppression to hide individuals within a group.

Definition 2.2.1. (k-*anonymity*) A dataset $\mathcal{D}$ satisfies k-anonymity if and only if each combination of quasi-identifiers in the dataset is present at least in k records. In other words, every individual is indistinguishable from at least $k-1$ others with respect to their quasi-identifiers, creating the so called *equivalence classes*.

The example in Table 2.1 presents a dataset that satisfies k-anonymity with $k=2$. Here, each combination of quasi-identifiers is shared by at least two records. As a result, the probability of correctly re-identifying a single individual is reduced, since an attacker can no longer establish a unique correspondence between quasi-identifiers and a person.

Name	Age	ZIP Code
Alessia	[20 - 29]	34***
Alessia	[20 - 29]	34***
Michele	[30 - 39]	35***
Michele	[30 - 39]	35***
Anna	[40 - 49]	36***
Anna	[40 - 49]	36***

Table 2.1: Example of k-anonymized data

To achieve k-anonymity in practice, it is required to first identify the quasi-identifiers and transform them in such a way that they are no longer useful for an attacker attempting re-identification [92]. In the previous example, some anonymization techniques were used, such as *generalization* and *suppression*.

First, **generalization** is a technique that consists of replacing a specific value for an attribute with a more generic one. An example is presented in Table 2.2. Here, the generalized attribute is the *Age* one, where the specific values are replaced with ranges that comprise the real value. Similarly, categorical attributes can be generalized by replacing detailed values with broader groupings, such as replacing the attribute *City* with the more aggregated attribute *Region*.
Through generalization, the distinctiveness of quasi-identifier combinations is reduced, thereby increasing the number of records that share the same values. This directly contributes to satisfying the k-anonymity requirement, since an attacker can no longer uniquely link a single individual to their record but only to a group of at least k individuals.

Name	Age	City
Alessia	22	Venezia
Alessia	27	Milano

(a) Original dataset

$\Longrightarrow$

Name	Age	Region
Alessia	[20-29]	Veneto
Alessia	[20-29]	Lombardia

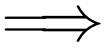
(b) Generalized dataset

Table 2.2: Example of data generalization technique

A key tool in data anonymization is the use of generalization hierarchies (also called taxonomies). A hierarchy defines increasingly coarse representations of an attribute's values, enabling flexible generalization while controlling the trade-off between privacy and data utility. For instance, the *Age* attribute may have a hierarchy ranging from exact values. e.g., 32, to intervals, such as $30-39$, to broader categories like Adult, while for *Location* values, a hierarchy may group ZIP codes into cities, then regions, and finally countries. During anonymization, values are replaced with higher-level categories depending on the privacy requirement, ensuring that individuals cannot be uniquely identified while still retaining meaningful aggregate information. Hierarchies thus provide a systematic way to transform sensitive attributes, balancing the level of privacy protection with analytical utility. Hierarchies, typically defined in advance by the data publisher, specify how attributes can be generalized to broader categories, guiding anonymization algorithms in balancing privacy and utility.
Another widely used anonymization technique is **suppression**. Suppression refers to the removal or masking of attribute values that could compromise privacy. In practice, this may involve completely eliminating an attribute from the dataset or partially obscuring its values. For instance, in Table 2.3, the attribute *ID* is suppressed entirely since it is an identifier, while *ZIP Code* has been partially masked by replacing some digits of the code with asterisks.
By reducing the precision or visibility of quasi-identifiers and identifiers, suppression decreases the risk of re-identification. Often, suppression is used in combination with generalization: while generalization broadens attribute values to form equivalence classes, i.e., a group of records that share the identical values for the quasi-identifier attributes, suppression ensures that highly unique or risky attributes do not leak information that could still single out individuals.
While k-anonymity has been highly influential as a foundational privacy model, it also suffers from several important limitations that reduce its effectiveness in practice. First,

Name	ID	ZIP Code
Michele	ABC1234	35133
Michele	ABC1235	35169

(a) Original dataset



Name	ZIP Code
Michele	35***
Michele	35***

(b) Suppressed dataset

Table 2.3: Example of data suppression technique

k-anonymity only guarantees that each record is indistinguishable from at least $k-1$ others with respect to quasi-identifiers, thereby preventing straightforward re-identification. However, it does not ensure protection against attribute disclosure. Consider the case in which all records within an equivalence class share the same value for a sensitive attribute, for example, a medical condition. In such a scenario, knowing that an individual belongs to that class is enough to reveal the sensitive information with certainty. Thus, even though identity disclosure is prevented, privacy is still compromised. This weakness was noted soon after the introduction of the privacy model [126].

Second, k-anonymity is vulnerable to homogeneity and background knowledge attacks. For example, in [86], the authors demonstrated that even when equivalence classes contain multiple sensitive values, attackers with external knowledge may still infer sensitive information with high confidence. In the case of a homogeneity attack, the sensitive attributes within an equivalence class lack sufficient diversity—for example, if nearly all individuals in the class share the same medical condition, membership in that class strongly suggests that condition. In a background knowledge attack, adversaries exploit external or auxiliary information to eliminate possibilities until only one sensitive value remains plausible, thereby re-identifying individuals despite the k-anonymity guarantee.

Third, achieving k-anonymity typically requires heavy generalization or suppression, which may substantially reduce the utility of the data. As highlighted in [117], increasing the value of k typically forces coarser generalizations, limiting the dataset's overall usefulness for analysis. This limitation illustrates the inherent privacy-utility trade-off, which becomes especially severe in high-dimensional datasets where the sparsity of unique attribute combinations makes it difficult to achieve anonymity without introducing excessive information loss [3]. In such cases, the dataset may remain privacy-preserving but lose much of its practical value for meaningful analysis.

Finally, k-anonymity provides no guarantee against composition attacks, in which multiple anonymized datasets are combined to re-identify individuals. Research has shown that if datasets anonymized separately under k-anonymity are linked together, equivalence classes may collapse, allowing re-identification with certainty [45].

Taken together, these limitations show that while k-anonymity has been a crucial foundation in privacy-preserving data publishing, it is widely regarded as necessary but insufficient. Its weaknesses have motivated the development of stronger models such as l-diversity [86] and t-closeness [80] which aim to mitigate specific vulnerabilities of the original framework and provide more robust protection against attribute disclosure.

l-diversity

The attacks described in [86] show that information leakage remains possible even when k-anonymity is enforced. Two attack scenarios were studied:

- *Homogeneity attacks* arise when all the individuals in the same equivalence class share the same value for an anonymized attribute. In this case, membership in the class directly reveals that sensitive value.
- *Background knowledge*, in which adversaries leverage external information, can help an attacker to narrow down the set of possible attributes within an equivalence class, increasing the re-identification risk.

To address these vulnerabilities, the l-diversity model has been proposed. In particular, it extends the k-anonymity model by requiring that, for each equivalence class, there must exist at least l distinct values for the sensitive attribute. The intuition is that diversity within sensitive attributes reduces the certainty with which an adversary can infer private information. Table 2.4 provides a simple illustration: after generalizing the attribute *Age*, an equivalence class is formed in which the sensitive attribute *Disease* contains $l = 2$ different values. As a result, even if an attacker can identify the equivalence class of a given individual, they cannot unambiguously determine the person's medical condition.

Name	Age	Disease
Tommaso	22	Lung cancer
Tommaso	27	Heart Disease

(a) Original dataset

Name	Age	Disease
Tommaso	[20 - 29[	Lung cancer
Tommaso	[20 - 29[	Heart Disease

(b) Anonymized dataset

Table 2.4: Example of l-diversity

This requirement mitigates the risks listed previously by significantly reducing the probability of an attacker inferring sensitive information with certainty.
However, l-diversity also has important weaknesses. First, it does not account for semantic similarities among the sensitive values. For instance, if the attribute *Disease* is generalized such that all patients with lung, skin, or breast cancer are grouped under the single label "cancer", then membership in that equivalence class still reveals with certainty that an individual has cancer, despite the presence of multiple distinct values. A second limitation with l-diversity derives from the possible distribution imbalance of the values of a sensitive attribute. For example, if 99% of individuals in the class share the same disease, then an attacker can still infer that value with a very high probability. This vulnerability is known as the *skewness attack*.

t-closeness

The limitations of l-diversity motivated the introduction of a stronger privacy model known as t-closeness [80]. The key idea behind this model is that the distribution of a sensitive attribute within each equivalence class should not differ "too much" from its distribution in the real dataset. In other words, the global distribution of sensitive

attributes is assumed to be publicly available information and equivalence classes are required to preserve this distribution up to a threshold t. By enforcing similarity between local and global distributions, t-closeness provides stronger protection against both homogeneity attacks (where all records in a group share the same sensitive value) and skewness attacks (where the distribution of sensitive values is highly imbalanced). The closer the class-level distribution is to the global one, the less information an adversary gains about an individual by learning that they belong to a specific equivalence class.
To measure the distance between the distribution of sensitive attributes in an equivalence class and their distribution in the overall dataset, statistical metrics such as the Earth Mover's Distance (EMD) [115] are commonly employed. The t parameter indicates a threshold for the distance between the distributions. Formally, an equivalence class is said to satisfy t-closeness if the distance between its sensitive attribute distribution and the global distribution is no greater than t. Similarly, a dataset is said to have t-closeness if it holds for all the equivalence classes.
However, t-closeness also has important limitations. From a practical perspective, enforcing distributional constraints often requires stronger generalization and suppression, which can lead to significant loss of data utility [42]. Furthermore, the model assumes that adversaries only rely on statistical distributions, and it may not provide sufficient protection against adversaries with richer background knowledge. Finally, determining an appropriate threshold t remains a major challenge: smaller values enhance privacy but severely compromise data utility, whereas larger values preserve utility but weaken protection.

Other Classical Techniques and Application

The classical privacy models of k-anonymity, l-diversity, and t-closeness rely primarily on generalization and suppression as mechanisms to achieve anonymity. While these approaches are effective in specific scenarios, they often lead to substantial information loss, especially in high-dimensional datasets where preserving privacy requires extensive transformations. As a result, a variety of additional anonymization techniques have been proposed within the field of statistical disclosure control (SDC) and privacy-preserving data publishing, aiming to balance privacy protection with the preservation of data utility.
Sampling is an anonymization technique in which only a subset of the original dataset is released. By reducing the number of available records, the probability of re-identification is lowered. Sampling is simple and often used in practice, but it reduces completeness and may bias the published data if not carefully designed [30]. However, naive random sampling can distort the statistical properties of the dataset and may fail to provide meaningful privacy guarantees. To address these issues, *β-sampling* was introduced as a more controlled probabilistic anonymization approach. In β-sampling, each record in the dataset is independently included in the published dataset with probability β ($0 < \beta < 1$). This process ensures that the adversary cannot determine with certainty whether a specific individual's record is present. The effectiveness of sampling can be further improved by accounting for population structure: techniques such as stratified or balanced sampling maintain the distribution of key subgroups while still providing privacy protection, thereby reducing bias and preserving the statistical utility of the

sampled dataset.

Perturbation techniques modify data values to mask individual records while preserving overall statistical properties. Common methods include adding random noise to numerical attributes, swapping values between records, or generating synthetic records that resemble the original data distribution. While perturbation offers flexibility and can preserve the utility of some analyses, making it suitable for many statistical or machine learning applications, it weakens the interpretability of individual records and may reduce the validity of fine-grained analyses that rely on precise attribute values, highlighting an inherent trade-off between privacy protection and data fidelity.

Together, these techniques illustrate the wide range of approaches available for anonymization beyond classical privacy models. Each method embodies trade-offs between privacy and data utility, and in practice, hybrid approaches combining generalization, suppression, sampling, and perturbation are often applied. Moreover, these limitations have fueled the exploration of alternative paradigms such as differential privacy, which aims to provide stronger, mathematically provable privacy guarantees beyond what conventional models can offer.

Tools for Anonymization

While privacy models such as k-anonymity, l-diversity, and t-closeness provide theoretical frameworks, their practical application requires software capable of handling real-world datasets and constraints. One of the most widely used tools for this purpose is ARX [106], an open-source anonymization framework that integrates classical and advanced techniques into a unified environment, allowing researchers and practitioners to implement privacy-preserving transformations efficiently.

In ARX, users can define generalization hierarchies for generalization, configure parameters for suppression, and apply privacy models such as k-anonymity, l-diversity, or t-closeness. The tool provides quantitative metrics to evaluate the trade-off between privacy protection and data utility, including re-identification risk estimates and information loss measures. Beyond generalization and suppression, ARX supports methods such as sampling and perturbation, which can be combined into hybrid anonymization strategies. This flexibility allows practitioners to iteratively test configurations, compare results, and select a solution that best fits the privacy requirements and analytical needs of a given use case.

Despite these capabilities, anonymization through ARX and similar frameworks remains constrained by the inherent trade-off between privacy and utility, particularly in high-dimensional datasets or scenarios with strong adversaries. These challenges have motivated the exploration of alternative paradigms such as synthetic data generation, which aims to preserve statistical patterns without directly exposing original records.

2.2.2 Synthetic Data Generation

Given the limitations of traditional anonymization techniques—specifically the degradation of data utility due to excessive generalization—research has shifted toward *Synthetic Data Generation* (SDG). Unlike anonymization, which modifies real records, SDG employs generative models to learn the underlying probability distribution $\mathbb{P}_{data}$ of the original dataset $\mathcal{D}$. Once learned, the model can sample entirely new data points $\mathcal{D}_{syn} \sim$

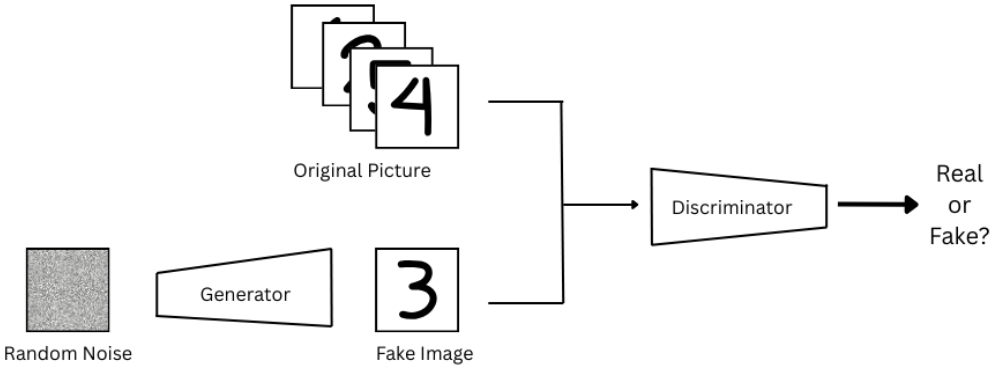


Figure 2.3: Graphical representation of the workflow of a Generative Adversarial Network.

$\mathbb{P}_{model}$ that statistically resemble the original data but do not map 1-to-1 to any real individual.
Modern synthetic data generation relies on four primary deep learning architectures: Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), Diffusion Models, and Transformers.

GAN

Generative Adversarial Networks (GANs) [52] consist of two competing neural networks: a generator (G) that synthesizes data from random noise z, and a discriminator (D) trained to distinguish these generated samples $\hat{x}$ from real training data x. Through this adversarial process, the generator iteratively learns to produce increasingly realistic samples to "fool" the discriminator. This interaction is formalized in the min–max loss function, where G aims to minimize the probability of D correctly identifying fake samples, and D aims to maximize it. The used loss function is called *min-max loss*, and is defined as:

$$\mathcal{L} = \mathbb{E}_x[\log(D(x))] + \mathbb{E}_z[\log(1 - D(G(z)))] \tag{2.6}$$

Figure 2.3 illustrates how the two networks interact. Specifically, GANs can be used to generate pictures when the neural network is implemented as a convolutional neural network. Alternatively, GANs can produce generic numeric data points when the networks are implemented as multi-layer perceptrons (MLPs).

VAE

Variational Autoencoders are another class of generative model composed of two networks: the *encoder* and the *decoder*. Unlike GANs, the interaction between the two networks is not adversarial. The encoder takes as input a data point x from the training set and encodes it into a low-dimensional latent space. The output of the encoder will be a 2-dimensional vector that identifies the two parameters of a multivariate Gaussian distribution, specifically, the mean $\mu_{z|x}$ and the covariance $\Sigma_{z|x}$. A latent vector z is then sampled from this distribution. The decoder is used to reconstruct the original data point

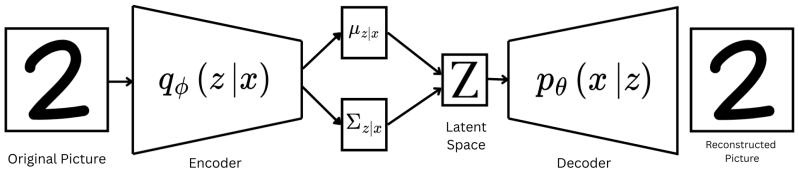


Figure 2.4: Graphical representation of the workflow of a Variational Autoencoder.

from z. The training of VAEs relies on a loss function that combines two components: a reconstruction loss, which encourages the decoder to accurately reproduce the input data, and a regularization term, typically the Kullback–Leibler (KL) divergence, which ensures that the learned latent space distribution remains close to a standard Gaussian.

$$\mathcal{L} = \mathbb{E}_z[\log p_\theta(x|z)] - D_{KL}[q_\phi(z|x)||p_\theta(z)] \tag{2.7}$$

The first term is the reconstruction term, while the second term is the Kullback-Leiber divergence term. By forcing the learned latent space distribution to be a standard Gaussian, the decoder can be used independently to generate new samples from random vectors drawn from a multivariate normal distribution. Figure 2.4 illustrates the architecture of a VAE, showing how the encoder maps input data to a latent space distribution and how the decoder reconstructs the data from samples drawn from this space. The figure highlights the flow of information during training and emphasizes the role of the latent space as a compact, structured representation of the input data.

In recent years, generative AI has evolved significantly, leading to the development of powerful models that surpass the performance of classical generative approaches. Among the most notable advancements are diffusion models and transformers, which have demonstrated remarkable capabilities in generating high-quality data across a variety of domains.

Diffusion Models

Diffusion models [59] have gained significant attention in the last years, particularly in the field of image generation. The core idea behind these models is to iteratively add noise to a data point and then learn to reverse this process, gradually denoising it to generate realistic samples. Figure 2.5 illustrates this forward and reverse process, highlighting how the model learns to reconstruct high-quality images from noisy inputs. The procedure begins with the forward process, in which the original image is gradually corrupted by adding noise over multiple steps until it becomes indistinguishable from random noise. Then, the backward process is performed: the model is trained to progressively remove the noise at each step, reconstructing the original data point from the

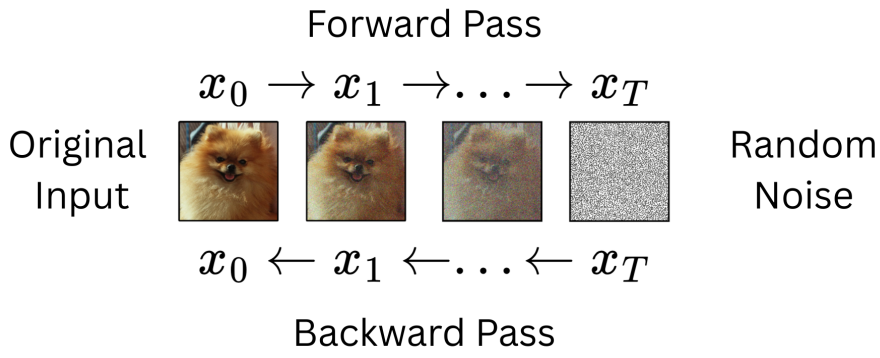


Figure 2.5: Forward and Backward Pass in the Diffusion Model for images (inspired by [48]).

corrupted version. During inference, only the backward process is used, allowing the model to generate new images starting from pure random noise and iteratively denoising them into realistic samples.

Transformers

Transformers, as discussed in Sec 2.1.2, gained popularity in the field of text translation, but they can be used for text generation. After the encoding of the text into tokens, the model is trained to predict the probability distribution of the next token in a sequence. Thanks to the attention mechanism, this distribution considers all previously generated tokens, allowing the model to capture long-range dependencies within the text. During inference, the model is given a prompt and generates text in an autoregressive manner, producing one token at a time according to the learned probability distribution.
These models are highlighted because they represent the current state of the art for data generation. While many other generative models exist, most are extensions or adaptations of the previously seen architectures.

2.2.3 Example of data generation architectures

Returning to tabular (or structured) data, many of the generative architectures discussed previously—such as GANs, VAEs, and transformers—have been adapted to generate synthetic structured datasets. In this section, we present some of the most widely used and well-known models for this purpose, highlighting their architectures and key mechanisms for generating realistic tabular data.

Conditional Tabular GAN

Conditional Tabular GAN (CTGAN) [137] is a generative adversarial network adapted for synthetic tabular data generation. In this particular model, categorical and numeric columns are treated in different ways. Consider a dataset with N rows and M columns, identified by the letters d and a with i and j as their indices respectively. The i-th row d_i is composed of n_{cat} categorical columns (or attributes) and n_{num} numeric columns. So the generic row will be:

$$d_i = \{x_1, x_2, \ldots, x_M\}, \ \ \forall i \text{ in } 1, \ldots N$$

where $n_{\text{cat}} + n_{\text{num}} = M$. The generic categorical value in d_i will be denoted as x_{c_j}, while the generic numeric value will be denoted as x_{n_j}.

Categorical attributes are encoded using a standard one-hot encoding. Numerical attributes, instead, are transformed with a technique called *mode-specific normalization*. In this approach, a Variational Gaussian Mixture Model (VGM) is fitted to each numerical attribute to estimate its modes and their corresponding distributions. Each value is then represented as a vector whose length equals the number of modes discovered by the VGM. The vector contains two pieces of information:

1. a one-hot encoded indicator of the most likely mode for that value
2. a scalar that specifies the relative position of the value within the selected mode

The encoded row is obtained by concatenating the one-hot vectors of categorical attributes with the mode-specific encoded vectors of numerical attributes. The resulting representation is then used to train the GAN as usual, with the key difference being the introduction of the *Conditional Generator*.
The conditional generator is designed to address the problem of *class imbalance*, i.e., the under-representation of rare attribute values in the training set. To solve this, the generator is conditioned on a specific categorical value k^*. In practice, this means that the generator explicitly incorporates k^* into its input, so that the distribution of generated rows includes sufficient samples with this rare attribute value. The conditional distribution of the generator for such rows can be formally written as:

$$\mathbb{P}(d_i) = \sum_{k \in C_j} \mathbb{P}_G(d_i | x_{c_j} = k^*) \mathbb{P}(x_{c_j} = k)$$

Where C_j the j-th categorical attribute. In this way, also low-represented values for categorical attributes will be present in the generated dataset.

Tabular Denoising Diffusion Probabilistic Model

Tabular Denoising Diffusion Probabilistic Model (TabDDPM) [72] extends diffusion-based generative modeling to the domain of structured tabular data. The idea is to adapt the general diffusion framework (introduced in Section 2.1.2) so that it can effectively handle the heterogeneous nature of tabular datasets, which typically combine both numerical and categorical attributes. The adaptation is conceptually straightforward but requires careful preprocessing and distinct diffusion strategies for different data types. For numerical attributes, the values are first transformed using the quantile transformer from Scikit-Learn [103]. This step maps arbitrary continuous distributions into a uniform or Gaussian-like distribution, making them more amenable to Gaussian diffusion processes. Categorical attributes, on the other hand, are encoded via one-hot encoding, which provides a simple yet effective way to represent discrete categories in a continuous vector space.

Once preprocessed, diffusion is applied separately for each type of attribute. Numerical columns are modeled using Gaussian diffusion [59], where noise is gradually added and then removed to reconstruct realistic values. Categorical columns instead rely on multinomial diffusion [60], a discrete variant of diffusion that perturbs and denoises categorical probability distributions instead of continuous values. This dual treatment ensures that both numerical variability and categorical semantics are properly captured. At inference time, the model starts from noise (Gaussian noise for numeric features, and uniform distributions for categorical ones) and iteratively denoises the data according to the learned process, ultimately generating synthetic rows that closely follow the original tabular distribution. By explicitly tailoring the diffusion mechanisms to the nature of each attribute type, TabDDPM achieves state-of-the-art performance in synthetic tabular data generation.

Realistic Relational and Tabular Transformer

Realistic Relational and Tabular Transformer (REalTabFormer or rtf) [122], is a transformer-based architecture designed to generate tabular data. The idea of using transformers with tabular data was proposed in [12] with the GReaT (Generation of Realistic Tabular data) model, which proposed using language modeling techniques to address the challenges of heterogeneous tabular structures.
The central idea is to represent tabular data as natural language text, thereby enabling the use of pre-trained large language models (LLMs). Each row of a table is converted into a sequence of sentences using the column names and their associated values. For example, a record with attributes *Age = 42* and *Occupation = Teacher* would be mapped to the sentences *"Age is 42"* and *"Occupation is Teacher"*. These sentences are concatenated to form a textual representation of the row. To avoid imposing artificial orderings between independent attributes, the sentences are randomly permuted, reflecting the fact that column order in tabular data typically carries no semantic meaning.
Once converted, the dataset is used to fine-tune the transformer model, which learns to generate realistic rows in this textual format. At inference time, the model samples new sequences that can then be parsed back into structured tabular data. This approach leverages the powerful contextual reasoning abilities of transformers, while bypassing the difficulties of directly modeling mixed data types (numeric and categorical) in their native tabular form.
REalTabFormer extends and improves on GReaT by addressing limitations related to data fidelity, relational consistency, and scalability. Its key improvements compared to GReaT are summarized in Table 2.5.
In the REalTabFormer model, the rows are transformed into text in a different way with respect to the GReaT model. While for categorical data no preprocess is done, apart from creating a unique vocabulary for each column and tokenization, the numeric attributes require careful treatment. If numeric values were simply cast to strings, the resulting vocabulary would be extremely large (covering every possible number), making training inefficient and limiting generalization.
To address this, REalTabFormer casts numeric attributes into fixed-length string segments, splitting each number into smaller tokens that represent only a few digits at a time. This transformation ensures that even large or highly precise numbers can be represented using a compact and reusable vocabulary. When numbers have fewer digits,

Improvement	GReaT	RTF
Fixed-vocabulary per column	Use a pretrained LLM on a corpus with unused words	GPT-2 trained on a fixed vocabulary for each column [97]
Support for relational data	Can be used only for single-table data	Can be used for relational datasets
Statistical stopping criterion	Does not account for it	the Q_δ statistic is used to detect overfitting and trigger early stopping during training
Target masking	The model can copy training data	It introduces masking of values to discourage copying training rows verbatim
Data Types	Numeric and Categorical	Numeric, Categorical, Datetime

Table 2.5: REalTabFormer and GReaT model Comparison

they are padded with zeros so that each value is represented consistently with the same number of tokens. An example transformation is shown below:

$$
\begin{gathered}
[1032.325345, 10.291, -3.0] \\
\downarrow \\
[\text{“10”}, \text{“32”}, \text{“.3”}, \text{“3”}] \\
[\text{“00”}, \text{“10”}, \text{“.2”}, \text{“9”}] \\
[\text{“-0”}, \text{“03”}, \text{“.0”}, \text{“0”}]
\end{gathered}
$$

So, for each numerical attribute, multiple tokenized columns are created. As written in the paper, this transformation reduces vocabulary size while preserving numeric precision. To conclude, datetime variables are transformed into numeric ones using the Unix timestamp representation, and then are treated as numeric variables.
The model is trained autoregressively: for each row, the transformer sequentially predicts the tokens of every column, sampling from the vocabulary specific to that attribute. This design ensures that the model learns both the semantic meaning of categorical variables and the fine-grained structure of numeric ones, while keeping the overall vocabulary compact and manageable.

Beyond the previously discussed models, other approaches for synthetic data generation include flow-matching methods [118], graph-based models [67], score-based techniques such as STaSy [71], Bayesian methods exemplified by PrivBayes [142], and statistical frameworks like synthpop in R [95].

2.3 Evaluation Metrics

While generative models provide powerful mechanisms to create synthetic datasets, their usefulness ultimately depends on how well the generated data preserves the properties of the original dataset while ensuring privacy. For this reason, evaluation metrics play a central role: they are used to assess the utility, by measuring how faithfully the synthetic data reproduces statistical patterns and supports downstream analyses, and the privacy, by estimating the risk of re-identification or information leakage. Establishing the right balance between these two dimensions is critical, as overly realistic synthetic data may compromise privacy, while overly private data may lack analytical value.

2.3.1 Fidelity and Utility Metrics

Evaluating generative models for tabular data is inherently complex because it involves a multi-objective trade-off. A perfect synthetic dataset must be statistically indistinguishable from the real data (*Fidelity*), useful for training downstream machine learning models (*Utility*), and secure against re-identification attacks (*Privacy*).

Fidelity Metrics

Fidelity metrics, often referred as broad utility or statistical fidelity, measure the degree to which a generated dataset captures the underlying statistical structure of the original data. Following a principled approach, these metrics are categorized into those assessing overall *Distribution Similarity* and those measuring *Specific Properties* [70].
Distribution similarity metrics evaluate the global resemblance between real and synthetic probability distributions, which is essential for capturing high-dimensional relationships that simpler tests might overlook. For instance, the *Maximum Mean Discrepancy* (MMD) [55] utilizes a kernel-based approach to compare mean embeddings in a Reproducing Kernel Hilbert Space, effectively quantifying the distance between two distributions. Similarly, the *Wasserstein Distance* provides a robust measure of similarity for numeric data by calculating the minimum cost of transforming one distribution into the other, while information-theoretic measures like *KL-Divergence* [73] and *Jensen-Shannon Divergence* [82] quantify the relative entropy or statistical distance between the synthetic and reference distributions. These multivariate methods are critical because they ensure that the synthetic data maintains the complex joint distributions necessary for reliable secondary research.
Conversely, measures for specific properties focus on individual components or local relationships within the dataset to verify that basic statistical characteristics are preserved. Univariate indicators are used to evaluate the goodness of fit for individual marginal distributions, such as the *Kolmogorov-Smirnov* (K-S) Test [87] for numeric columns and the *Chi-Squared* Test for categorical variables, which verify that the frequency and range of specific attributes remain consistent. Beyond individual marginals, bivariate properties are assessed through metrics like *Pairwise Correlation Difference*, which calculates the variance between the correlation matrices of the real and synthetic datasets to ensure that linear relationships and dependencies between variables are not lost during the synthesis process.

While these fidelity assessments are foundational, the current landscape still lacks a unified nomenclature, with researchers often employing a vast variety of metrics across these categories without a standardized protocol. It is also noted that high fidelity does not inherently guarantee privacy; in fact, similarity-based metrics like distance to real data can be at odds with privacy objectives, as a synthetic record that perfectly mimics a real patient for fidelity reasons may inadvertently facilitate re-identification.

Utility Metrics

Utility metrics represent all those metrics that measure the power of a dataset to be used for training other models for downstream tasks. For example, consider a scenario where a researcher wants to study the effect of a medicine but cannot access real data due to privacy or cost constraints. In this case, a synthetic dataset can be used as a substitute. However, for the synthetic data to be useful, it must preserve the relationships between the response variable and the predictors present in the real dataset. In other words, a model trained on the synthetic data should produce results comparable to a model trained on the original data. This ability to retain meaningful patterns and relationships is what defines the utility of synthetic data.

To evaluate utility, the commonly used methodology is *Train on Synthetic, Test on Real* (TSTR) [37], or *model compatibility* [100], which is typically compared to *Train on Real, Test on Real* (TRTR). In this framework, a model is trained on the synthetic dataset and then evaluated on the real dataset. If the performance metrics differ significantly between TSTR and TRTR, it indicates that the synthetic dataset has low utility. Conversely, if the results are similar, the synthetic data effectively preserves the relationships in the original data and is considered highly useful for downstream tasks.

Another metric used to assess utility and similarity is the *Machine Learning Efficacy*, or *Discriminator* [85]. This approach involves training a classifier, often XGBoost, to distinguish between real and synthetic records. If the classifier achieves a prediction accuracy close to 100% or 0%*, it indicates that the synthetic records are easily distinguishable from the real ones, suggesting that the synthetic data distribution differs substantially from the original. Conversely, if the classifier's accuracy is near 50%, it implies that the model cannot reliably tell real and synthetic records apart, indicating a high degree of similarity between the two datasets.

It is important to note that these metrics represent only a subset of the tools available to evaluate synthetic data. No single metric can definitively determine the quality of a synthetic data generator. Instead, by combining multiple metrics, such as fidelity, utility, and machine learning efficacy, researchers can obtain a more comprehensive understanding of how well the synthetic data preserves the properties and relationships of the original dataset.

*An accuracy of 0% simply indicates that the classifier is systematically inverting the labels, classifying real records as synthetic and vice versa. In the context of this binary classification task, such inversion does not affect the assessment of distinguishability.

2.3.2 Privacy Metrics

Regarding evaluation, privacy metrics for synthetic data can be broadly classified into two categories:

- Similarity-based Metrics.
- Attack-based Metrics.

On one hand, similarity-based metrics are designed to ensure that the synthetic data is sufficiently different from the original dataset to prevent privacy leaks. For instance, if a synthetic dataset were an exact copy of the training data, it could not be safely released, as it would expose real personal or sensitive information. These metrics rely on a similarity measure to compare the synthetic and real data, along with a predefined threshold that indicates when the generated data is too close to the original. An example of such a metric is the *Identical Match Share* (IMS) which computes the percentage of rows in the synthetic dataset that are exact copies of the rows in the training set. Other similarity-based metrics will be discussed in the following sections.
On the other hand, attack-based metrics simulate potential privacy breaches to evaluate the risk associated with releasing synthetic data. In this approach, an attacker (or threat model) uses the synthetic dataset to attempt to infer sensitive information about individuals in the original data. The effectiveness of these attacks on a set of real records serves as a measure of privacy leakage. It is crucial to define at the beginning the level of knowledge the attacker has access to. For example, in the *black-box attack*, the attacker will be able to access only the information provided by the generated dataset itself. In contrast, in the *white-box attacks*, the attacker is also allowed to access the generating process information; so, the model's architecture, the hyperparameters, or even the weights of the model. There are also intermediate scenarios in which the attacker has partial knowledge of the system, capturing a spectrum of realistic threat models.
While the primary incentive for using synthetic data is to facilitate privacy-preserving data sharing, research indicates that many studies assume inherent privacy benefits without empirical verification. To rigorously quantify these risks, the literature increasingly focuses on two primary attack-based dimensions: *Membership Inference Attacks* (MIA) and *Attribute Inference Attacks* (AIA) [70].
Membership Inference assesses the risk of an adversary determining whether a specific individual's record was part of the original training dataset. This is a fundamental privacy concern because confirming an individual's presence in a specific cohort can reveal sensitive information. MIAs are commonly evaluated using methods like record matching, which checks for identical or near-identical records between synthetic and real sets, and hold-out set distinguishing, where a classifier attempts to tell the difference between training records and those from a separate hold-out set. Research by Shokri et al. [121] established the foundational framework for these attacks, demonstrating that the tendency of machine learning models to overfit on their training data allows adversaries to recognize membership with high confidence.
Attribute Inference addresses the risk of an intruder deducing the value of sensitive attributes for a known individual by having access to the synthetic data. This type of disclosure occurs when an attacker can leverage the correlations captured by the generative model to predict private information about a specific data subject. Common evaluation techniques for AIAs involve inference based on classification or regression models,

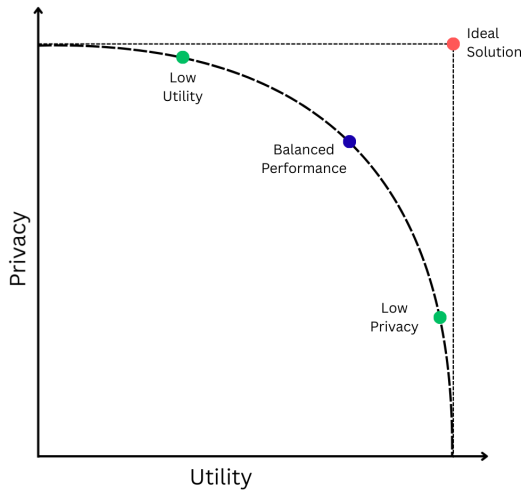


Figure 2.6: Privacy-Utility tradeoff.

where a predictive model is trained on the synthetic data to determine how accurately it can predict sensitive variables in the real dataset. As highlighted by Kaabachi et al. [70], although AIA represents a direct threat to the confidentiality of patient data, it is significantly less represented in current evaluation literature compared to MIA, appearing in only a small fraction of analyzed instances.
It is crucial to recognize that traditional similarity-based metrics, such as the distance to the closest record, are often inadequate for ensuring true privacy. Ganev and De Cristofaro [44] have demonstrated that successful inference attacks can occur even when the synthetic data appears statistically dissimilar from the original dataset. Consequently, a robust privacy assessment requires moving beyond simple resemblance tests toward principled adversarial auditing that explicitly evaluates resilience against both membership and attribute disclosure.

2.3.3 The Privacy-Utility Tradeoff

The previous discussions on utility, fidelity, and privacy metrics have highlighted an important connection between them. For instance, if a SDG method is able to generate a dataset with perfect utility and fidelity, it can create problems on the privacy side. Consider the extreme scenario in which the generated dataset is an exact copy of the training data: while utility and fidelity scores would be maximal, the privacy risk would be at its highest, as all sensitive information is exposed. In the other extreme-case scenario, the SDG produces a dataset that is completely different from the training data. In this case, all the privacy scores would have a perfect score but utility and fidelity scores would be very low. From this consideration, we can identify a tradeoff between utility, in the broader sense[†], and privacy, which is depicted in Figure 2.6.
Technologies that provide a high privacy protection have also severely degraded utility. On the other side, a good utility corresponds to a low level of privacy protection. The

[†]Considering both Fidelity and Utility

current research is focused on creating models or methodologies that maximize the acceptable threshold. Using anonymization techniques is the best for privacy protection as seen in Section 2.2.1, but, due to the modifications the dataset undergoes, the utility and the fidelity score are low or cannot be computed properly. Regarding SDG, the methodology can achieve a better balance; however, improper tuning or insufficient regularization may lead to privacy risks if the model overfits, effectively memorizing and reproducing training records. Conversely, overly constrained or undertrained models may preserve privacy but fail to capture the complex relationships between attributes, resulting in low utility and poor fidelity.

The goal of current research is not merely to move along this curve, but to *shift the frontier itself*, enabling higher utility for a fixed privacy budget. While standard generative models (like vanilla GANs) can achieve a better balance than traditional anonymization, they lack stability. Without explicit constraints, a GAN may drift toward the overfitting extreme, memorizing training samples to minimize its loss function.
Furthermore, relying solely on *post-hoc* attack metrics (like checking for re-identification after generation) is risky. It does not provide a formal safety guarantee. To address this, we require a framework that offers a mathematically provable bound on privacy leakage *during* the training process. This necessity leads us to the concept of *Differential Privacy*.

2.4 Differential Privacy for Tabular Data Generation

Now that the legal concept of privacy has been introduced and the main techniques to protect datasets prior to release, namely anonymization and synthetic data generation (SDG), have been discussed, the final method to mitigate privacy risks can be addressed: *Differential Privacy* (DP) [32]. Before presenting a formal definition of DP, it is essential to introduce some foundational concepts and definitions that underpin the framework.

Let $\mathcal{D}$ denote a generic dataset. A dataset $\mathcal{D}'$ is called a *neighboring dataset* of $\mathcal{D}$ if and only if it differs from $\mathcal{D}$ by exactly one element or row. In practice, two datasets are neighbors if they differ only in the presence or absence of one individual.

Definition 2.4.1. (*Differential Privacy*) Given two neighboring datasets $\mathcal{D}$, $\mathcal{D}'$, given a randomized function $\mathcal{K}$, and given $S \subseteq Range(\mathcal{K})$; then, $\mathcal{K}$ is said to satisfy ε-differential privacy if the following inequality holds.

$$\mathbb{P}[\mathcal{K}(\mathcal{D}) \in S] \leq \exp(\varepsilon) \cdot \mathbb{P}[\mathcal{K}(\mathcal{D}') \in S] \tag{2.8}$$

So, this definition states that, if ε-differential privacy holds, then the presence or absence of an individual from the dataset will affect the result of the function $\mathcal{K}$ by a value that depends on the ε parameter, also known as *privacy budget*. In other words, ε-differential privacy limits the influence of every individual in the given dataset $\mathcal{D}$. Conceptually, this guarantees plausible deniability [11]: an attacker cannot confidently determine whether a specific individual's data was used in the training set. This interpretation is crucial on the attacker side; in fact, if a function is ε-differentially private, then the attacker cannot infer whether a specific individual's data was used during training, making MIAs much harder to perform. By ensuring that individual contributions are negligible, differential

privacy effectively reduces the risk of targeting specific records. In [33], the concept of differential privacy was extended by adding a shift for the case of unlikely events. We refer to this new mechanism as (ε, δ) - differential privacy; in particular, the following inequality must hold.

$$\mathbb{P}[\mathcal{K}(\mathcal{D}) \in S] \leq \exp(\varepsilon) \cdot \mathbb{P}[\mathcal{K}(\mathcal{D}') \in S] + \delta \tag{2.9}$$

The privacy budget ε and the shift parameter δ must be set by the user and lie within specific domains: $\varepsilon > 0$ and $0 < \delta < 1$. A low value of ε, such as 0.1 or 1.0, corresponds to strong privacy guarantees, whereas values above 10.0 indicate very weak privacy protection. The parameter δ accounts for rare events where the mechanism may fail to provide strict privacy, meaning that there could exist outcomes that completely reveal private information; however, the probability of such outcomes is at most δ. In practice, a common rule of thumb is to set $\delta < \frac{1}{N}$, where N is the number of data points. For highly sensitive datasets, a stricter setting of $\delta < \frac{1}{N^2}$ can be used to provide stronger privacy guarantees.
To summarize, the δ parameter addresses the following aspects:

- Handles extremely unlikely events: Allows a very small probability of unlikely outcomes that could reveal private information, making the mechanism more practical than strict ε-DP
- Balances privacy and utility: Strict ε-DP can require adding a lot of noise, which may severely degrade the utility of the data or model. Allowing a tiny probability δ of failure lets designers use less noise overall while still providing strong privacy for the overwhelming majority of cases.
- Captures realistic adversarial assumptions: In practice, attackers may only occasionally have extreme capabilities or encounter rare outputs. The δ parameter formalizes this by acknowledging that absolute guarantees are impossible in these edge cases but bounding their probability.
- Enable composition as a DP property which is crucial for the mathematical framework, allowing the creation of complex DP systems.

The mathematical formulation of Differential Privacy provides a formal guarantee of *plausible deniability* [11]. This concept implies that because the output of the mechanism $\mathcal{K}$ is almost equally likely to have occurred whether or not a specific individual's data was included in $\mathcal{D}$, no observer can definitively prove an individual's participation. Formally, for any outcome $\mathcal{S}$, the ratio of probabilities $\mathbb{P}[\mathcal{K}(\mathcal{D}) \in \mathcal{S}]/\mathbb{P}[\mathcal{K}(\mathcal{D}') \in \mathcal{S}]$ is bounded by $\exp(\varepsilon)$, ensuring that an individual can always claim their data was not used, as the output is statistically consistent with both hypotheses.

While standard DP protects a single individual, many scenarios require protecting groups of individuals (e.g., families in a healthcare dataset). This is addressed through *Group Differential Privacy*[23].

Definition 2.4.2. A mechanism $\mathcal{K}$ satisfies k-Group Differential Privacy if for any two datasets $\mathcal{D}$ and $\mathcal{D}^{(k)}$ that differ by at most k records, the following holds:

Property	k-Anonymity	Differential Privacy	Plausible Deniability
Scope	Dataset Property	Algorithm Property	Record/Mechanism Property
Guarantee	Indistinguishability in group	Output Stability	Indistinguishability of Input
Vulnerability	Homogeneity Attack	Utility Loss	Parameter Tuning

Table 2.6: Comparison of key properties: k-Anonymity, Differential Privacy, and Plausible Deniability.

$$\mathbb{P}[\mathcal{K}(\mathcal{D}) \in \mathcal{S}] \leq \exp(k\varepsilon) \cdot \mathbb{P}[\mathcal{K}(\mathcal{D}^{(k)}) \in \mathcal{S}] \tag{2.10}$$

This property follows naturally from the standard definition: if a mechanism is ε-DP, it is automatically $k\varepsilon$-DP for a group of size k. However, as k increases, the privacy guarantee degrades linearly, requiring a very small ε to maintain strong protection for large groups.

Building upon the scaling properties of group privacy, the conceptual and mathematical links between Differential Privacy, traditional privacy models, and plausible deniability define the transition from heuristic, data-centric protection to rigorous, algorithmic guarantees. At the core of the Differential Privacy framework is the formalization of plausible deniability, a property ensuring that the output of a mechanism is statistically nearly as likely to occur whether or not a specific individual is included in the dataset. This provides a mathematical "safety net" where an individual can always claim their data was not used, as the results remain consistent with both hypotheses. While standard individual privacy focuses on the single data subject, the Group Differential Privacy property extends this deniability to a set of records; by bounding the influence of any k individuals, the framework ensures that even small groups maintain a level of protection against being singled out. This framework fundamentally differs from traditional privacy models like k-anonymity, which provides a property of the dataset itself—ensuring individuals are indistinguishable from a "crowd" of others based on quasi-identifiers. Unlike k-anonymity, which often relies on assumptions regarding limited background knowledge and is vulnerable to the "curse of dimensionality", Differential Privacy is designed to be resistant to adversaries with arbitrary background knowledge. It effectively moves the goalpost from hiding identities within a group to strictly hiding the influence of any individual on the final output. While traditional models like l-diversity or t-closeness attempt to prevent membership or attribute inference by generalizing or suppressing data, Differential Privacy achieves these goals through the controlled injection of noise. This offers a provable upper bound on privacy risk that traditional heuristic models cannot provide, addressing the "Groundhog Day" [124] of recurring re-identification vulnerabilities found in classical anonymization.
The following table highlights the key differences between the presented concepts.

2.4.1 Learning under Differential Privacy

The definitions alone are not sufficient to deal with the problem of learning functions using DP mechanisms. Initial studies on DP properties, such as those in [53], focused on **composability**. This property allows the calculation of the cumulative ε and δ when multiple DP mechanisms are applied to the same dataset, or when one DP mechanism

operates on the output of another. As highlighted in [2], composability enables the creation of a privacy accountant that tracks the total privacy loss during the execution of composed functions, often referred to as the *privacy cost*. This is crucial for designing algorithms that perform multiple DP queries while ensuring overall privacy guarantees.

Another fundamental property, discussed in [35], is the **post-proccess immunity**. This property states that, for any (ε, δ) - differentially private function $\mathcal{K}$, and for an arbitrary randomized mapping f, the composition $f \circ \mathcal{K}$ is still a (ε, δ) - differentially private function. In other words, no post-processing of the output of a DP mechanism can weaken the privacy guarantees already provided by $\mathcal{K}$.

One last key property is the **privacy amplification by subsampling** [6]. It has been observed that, to increase the privacy protection of a function, it is sufficient to apply random sampling to get a new dataset from the original one. Intuitively, the chances of leaking information of a specific individual decrease because there is a chance that the individual is not included in the sampled subset.

These three properties are fundamental for implementing differential privacy in stochastic gradient descent for deep learning, leading to the DP-SGD method proposed in [2]. DP-SGD modifies the gradient computation of standard SGD by clipping gradients at each step and adding random noise before updating the model weights. The type of noise used defines the privacy mechanism. The Laplacian Mechanism [34] adds Laplace-distributed noise calibrated to the privacy budget and is suitable for ε-differentially private functions. The Gaussian Mechanism [35] adds Gaussian noise and is compatible with (ε, δ)-differentially private functions, leveraging the relaxation term δ for practical applications.

It is important to emphasize several aspects of the DP-SGD procedure. First, DP-SGD satisfies the post-processing immunity property, meaning that any further transformation applied to the trained model does not compromise its (ε, δ)-differential privacy guarantees. Second, the use of mini-batches during the stochastic gradient descent procedure leads to privacy amplification by subsampling, as each individual record has a reduced probability of being included in any given batch, thus further limiting the potential privacy leak. Finally, the SGD procedure can be seen as a composition of multiple functions, and the composability property ensures that the total privacy budget for the entire training process can be accurately computed. To achieve this, a privacy accountant is employed to track the cumulative privacy cost across all gradient updates, ensuring that the model's overall privacy guarantees remain valid throughout training.

The privacy budget has a direct impact on the training of a machine learning model under DP-SGD. A smaller privacy budget (lower ε) corresponds to stronger privacy guarantees, but also requires the injection of more noise into the gradients at each training step, which can degrade model performance and slow convergence. Conversely, a larger ε reduces the amount of noise and allows the model to better fit the data, but weakens privacy protection. The mechanisms that control this balance are the noise injection mechanism and the privacy accountant. The noise injection mechanism determines the magnitude of noise to be added to the gradients at each step to satisfy the target differential privacy parameters. Meanwhile, the privacy accountant uses the composability property to track the cumulative privacy loss over multiple training steps, ensuring that the overall (ε, δ) budget is not exceeded. Among the most widely used privacy accountants are the Moment Accountant [2] and the Rényi Differential Privacy (RDP)

Accountant [135]. The RDP Accountant is based on the concept of Rényi differential privacy [93], which states that a function f is (ε, α)-Rényi differentially private if, for any two neighboring datasets $\mathcal{D}$ and $\mathcal{D}'$, the following inequality holds:

$$D_\alpha(f(\mathcal{D})||f(\mathcal{D}')) \leq \varepsilon \tag{2.11}$$

where $D_\alpha(\cdot||\cdot)$ is the Rényi divergence that is computed as follows.

$$D_\alpha(P||Q) = \frac{1}{\alpha - 1} \log \mathbb{E}_{x \sim Q} \left[\frac{P(x)}{Q(x)} \right]^\alpha \tag{2.12}$$

In practice, this second type of accountant, the RDP Accountant, and its corresponding definition of Rényi differential privacy are preferred because:

1. RDP composes linearly:
2. While (ε, δ)-differential privacy allow for small privacy breaches due to the relaxation term, RDP does not allow for it even with weak parameters because here is always some uncertainty left in the adversary's inference
3. It gives closed-form privacy loss bounds for Gaussian noise, subsampling, and other operations, which is not possible with the gaussian accountant
4. Possibility to convert back to (ε, δ)-differential privacy parameters

These theoretical concepts, while fundamental, can be challenging to implement correctly in practice when building differentially private machine learning models. To facilitate this process, frameworks such as Opacus [140] have been developed. Opacus acts as a wrapper around standard PyTorch [102] models, handling the computation and injection of noise at each training step, as well as tracking the cumulative privacy loss through a Renyì Differential Privacy accountant. By automating these tasks, Opacus simplifies the deployment of DP-ML models while ensuring that the specified (ε, δ) privacy guarantees are maintained throughout training.

2.4.2 Implementations

Differential privacy has been applied in many domains throughout the years. The most common application is in medical problems, as discussed in [146]. In this paper, the Gaussian DP has been applied to an image classification problem, highlighting the need for models that achieve high accuracy while protecting sensitive patient information in the training data. Another important application is in recommended systems [17], in which the private information of each user can be leaked or inferred by attackers [18, 68]. To mitigate such risks, many methods implementing DP were developed as protections from those attacks [78, 112, 145]. DP has also been integrated into *Federated Learning* (FL) [79], where it has been proven that MIAs are particularly effective [65]. Consequently, several DP-enhanced FL methods have been developed to improve privacy protection in this context [46, 90]. Differential Privacy has also been applied to synthetic data generators, addressing some of their inherent privacy weaknesses. In particular, it has been shown that Property Inference Attacks can be particularly effective

against GAN-based generators [144]. Similarly, Model Inversion Attacks [41] can use model's output, and parameters if available, to infer sensitive features of the training data. Notably, deep neural methods are particularly susceptible to leak information under these types of attacks [143]. Despite that, previously presented attacks are evaluated on generic SGD, the discussion of privacy leak risks can be extended to synthetic tabular data generators [123]. In the following sections, some differentially private tabular data synthesizers are presented.

DPCTGAN [40], is the natural extension of CTGAN using DP. In particular, the DP mechanism is applied exclusively to the Discriminator, or Critic. This design choice serves two purposes: first, applying DP to both the Generator and Discriminator can create convergence issues during training; second, the Discriminator is the only network that directly interacts with the real data, while the Generator learns indirectly through the Discriminator's feedback. By restricting the privacy mechanism to the Discriminator, DPCTGAN effectively protects sensitive information in the training data while maintaining stable training dynamics for the Generator.

PATE-GAN [69] leverages the *Private Aggregation of Teacher Ensembles* (PATE) framework. Instead of a single Discriminator, it partitions the data into k disjoint subsets and trains k separate "Teacher" discriminators.

- The Teachers vote on whether a generated sample is real or fake.
- The votes are aggregated with Laplacian noise.
- The "Student" (Generator) trains using only this noisy aggregate vote.

This ensures that the Generator's learning signal is differentially private with respect to the original data partitions.

AIM [89] is a novel algorithm designed for generating differentially private synthetic data. It follows the well-established select–measure–generate paradigm:

1. Selection: choosing queries, often low-dimensional marginals[‡], from a larger set.
2. Measurement: adding noise to the selected queries to enforce differential privacy.
3. Query Refinement: identifying and retaining the queries that most accurately explain the noisy measurements.

What makes AIM novel is its iterative and adaptive workflow. Rather than performing selection and measurement just once, AIM repeats these steps over multiple iterations. At each iteration, the mechanism identifies queries that are poorly approximated by the current synthetic dataset and prioritizes them in subsequent measurements. This adaptive approach progressively enhances the quality of the selected queries, ensuring that the synthetic data better captures the underlying structure of the original data. AIM also provides analytic error bounds, allowing users to estimate the expected accuracy

[‡]Desired statistical properties of the real dataset such as totals and conditional counts

of the synthetic dataset for each query. Its modular design supports a wide range of query types, including counts, marginals, and more complex statistics, making it suitable for high-dimensional datasets. By focusing the privacy budget on the most informative queries and using iterative refinement, AIM achieves a superior balance between privacy and utility compared to one-shot mechanisms. The method is particularly effective when the downstream analysis tasks or workload queries are known, as it tailors the synthetic data to meet these specific needs.

Despite the many formulations of Differential Privacy, each provides strong privacy guarantees grounded in rigorous mathematical definitions. To achieve DP in practice, noise injection plays a central role. Interestingly, anonymization techniques can also be interpreted as introducing noise into the data, which in some cases makes them compatible with DP. For instance, in [81], it was shown that applying specific k-anonymization methods preceded by β-sampling results in a weaker but still meaningful guarantee, namely $(\beta, \varepsilon, \delta)$-differential privacy. This observation highlights the fundamental connections between anonymization, synthetic data generation, and differential privacy, and sets the stage for exploring integrated approaches that aim to achieve stronger privacy guarantees while preserving data utility.

Chapter 3

Attack-based Metrics Framework

3.1 Introduction

As previously explained in Section 2.2.2, Synthetic Data Generation (SDG) has become one of the most fundamental techniques as Privacy Enhancing Technology (PET). Unfortunately, SDG is far from perfect [129], in fact, there are many challenges that remain to be addressed.

To start with, one crucial challenge lies in trustworthiness and quality measurement. It is often difficult to determine how much we can rely on analyses or models trained on synthetic data, since generative methods may introduce subtle distortions or biases. Current evaluation metrics are limited: some fail to capture the diversity or fidelity of the original data, while others are hard to interpret or relate to downstream performance.

A second important issue is the trade-off between privacy and utility. Strong privacy protections, such as differential privacy, can degrade the usefulness of the generated data by injecting too much noise. Moreover, privacy guarantees that hold today may weaken in the future as attacker models evolve or new auxiliary data become available.

Synthetic data also struggles to represent low-density regions and minority groups. Generative models tend to perform poorly in regions where original data is scarce, meaning that outliers, rare conditions, or underrepresented populations may be poorly captured. This is often a result of the model's objective function (e.g., in Generative Adversarial Networks), which prioritizes learning the dominant modes of the data distribution to "fool" the discriminator, thereby neglecting low-frequency samples. This limitation not only affects fairness but can lead to biased or incomplete downstream results.

To conclude, ensuring fairness in synthetic data is also non-trivial. A dataset may appear balanced, yet still lead to unfair model behavior when applied to real data. Furthermore, multiple and sometimes conflicting definitions of fairness make it difficult to decide which to enforce, and these constraints can further reduce data utility.

Given these limitations, it is critical to evaluate synthetic data in terms of both privacy and utility/fidelity to ensure that releasing a generated dataset does not violate individual privacy. In Section 2.3, the first distinction between the privacy metrics is made. On one hand, Similarity-based metrics rely on the observation that, if a generated dataset is too similar (according to a pre-defined metric) to the real one, some privacy breaches may occur. On the other hand, Attack-based metrics assume the existence of an attacker,

or threat model, who uses information obtained from the synthetic dataset to infer private information of real individuals.

This chapter presents a state-of-the-art framework for attack-based metrics, namely *Anonymeter* [49]. Furthermore, we introduce a novel privacy evaluation method that leverages Contrastive Learning (CL) to map tabular data into a continuous embedding space. This approach enhances the sensitivity of the Singling Out attack by capturing semantic similarities that standard predicate-based searches may miss, as originally presented in our work [99]. The main contributions are:

- **Contrastive Learning Enhancement**: We propose a novel extension to the Anonymeter framework that utilizes Contrastive Learning (CL) to map tabular records into a semantic embedding space, enabling the detection of complex, multi-attribute vulnerabilities.
- **Scalable Singling Out**: We develop a density-based Singling Out attack using Local Outlier Factor (LOF) within the learned embedding space. This method significantly improves computational efficiency on high-dimensional datasets compared to traditional brute-force predicate searches.
- **Evaluation of Embedding-Based Metrics**: We formally evaluate the application of contrastive embeddings to Linkability and Inference attacks, identifying critical trade-offs between the marginal statistical gains and the high computational costs required for these complex attack vectors.
- **Critique of Similarity Metrics**: We provide an empirical analysis of the Distance to Closest Record (DCR) metric, demonstrating that while it correlates with risk, enhancing it with learned embeddings (CL+DCR) does not yield additional sensitivity, validating the use of standard Euclidean metrics for this specific task.

3.2 Anonymeter

Anonymeter [49] is a statistical framework designed to compute the risk of three different privacy attacks that use synthetic data as prior knowledge. These three attacks, namely Singling Out, Linkability, and Inference, are highlighted in guidance from the Article 29 Working Party (WP29) [5], the predecessor to the European Data Protection Board, as key risks to evaluate when assessing the effectiveness of an anonymization technique.
The framework's high-level process is depicted in Figure 3.1. Anonymeter evaluates privacy risk by comparing the success of an attack against two distinct populations. First, the attack is run against the real dataset (the train set), which was used to train the synthetic data generator. Second, the same attack is run against a hold-out set (the control set), which consists of real records that were not used during the generator's training. This comparison is crucial: a successful attack on the train set that fails on the control set suggests that the generator has "memorized" and leaked specific information about the training data, rather than learning the general patterns shared by both sets. This holds true because the train-control split, conducted through random sampling, ensures

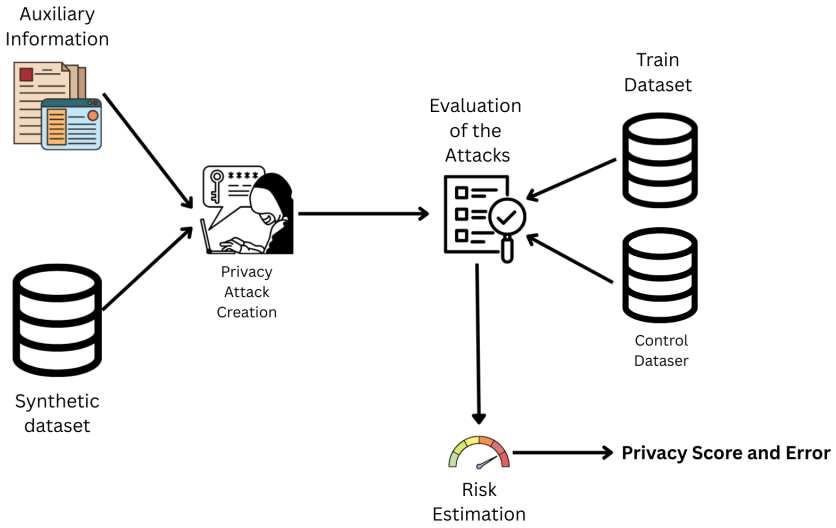


Figure 3.1: Anonymeter Framework scheme (this picture is inspired by [49])

statistical similarity between the two sets. A "naive" random attack is also performed as a baseline to ensure that the learned attack provides a meaningful advantage.
In this framework, the attacker leverages the synthetic data as prior knowledge to launch the attacks. For the Inference attack, this is supplemented with additional *auxiliary information*. The created attacks are tested on both the training and control datasets, and the resulting privacy risk R is computed as:

$$R = \frac{r_{\text{train}} - r_{\text{control}}}{1 - r_{\text{control}}} \tag{3.1}$$

This formulation normalizes the risk by the attack's baseline success on the hold-out set. The numerator measures the excess success the attacker achieves by targeting the training data (which the synthetic data may have memorized) versus the unseen control data. The denominator, instead, measures the maximum possible excess and acts as a normalizing factor. This normalization makes the final risk measure more realistic, considering the attack on unseen records and isolating the privacy breach specifically caused by the generator's potential memorization. If both risk rates are equal, then the overall risk measure is 0, so the access to synthetic data provides no advantage to the attacker. Alternatively, if the $r_{\text{train}} = 0.8$ and $r_{\text{control}} = 0.6$, then $R = 0.5$, indicating that the synthetic data is responsible for 50% of the possible privacy risk (beyond the baseline).
In the "Risk Quantification Phase", the attacks are tested on both the training and control set. Each individual attack attempt (e.g., on a single record d_i) has a binary outcome, o_i (success or failure), which is interpreted as a Bernoulli random variable. Assuming each attack attempt is independent, the entire set of attacks on a dataset constitutes a series of Bernoulli trials. Then, the estimated risk for an attack is the probability of successes for the attacker in all these attempts and is computed as follows:

$$r = \frac{N_S + z_\alpha^2/2}{N_A + z_\alpha^2} \tag{3.2}$$

Where N_A denotes the total number of performed attacks, N_S is the number of successful attacks, and z_α is the probit function corresponding to a level of confidence α.
To account for uncertainties, the confidence interval for these metrics is computed. In particular, the radii of the confidence interval for the risk are computed using the Wilson Score Interval [136] as:

$$\delta_r = \frac{z_\alpha}{N_A + z_\alpha^2} \sqrt{\frac{N_S(N_A - N_S)}{N_A} + \frac{z_\alpha^2}{4}} \tag{3.3}$$

Having established the statistical framework for risk quantification, we now detail the three specific attacks that Anonymeter implements.

Singling Out

Singling Out attacks aim to isolate a unique individual within a dataset using a set of attributes. This is performed by generating *predicates*—logical rules formed by attribute-value pairs. A record is considered "vulnerable" if a predicate uniquely identifies it. For example, in a census dataset, the predicate:

$$\{\text{Occupation} == \text{Professor}\} \wedge \{\text{Age} \geq 70\}$$

might identify a single unique individual.
Anonymeter creates those predicates by analyzing the synthetic data, assuming that if the generator memorized a unique record, that record's unique properties will be present in the synthetic data. Specifically, the predicates are created by identifying the extreme values or rare combinations within the synthetic data. For example, computing the minimum and maximum values for numeric attributes, the predicates can be:

$$\{a_{\text{numeric}} \geq \max \mathcal{D}[a_{\text{numeric}}]\}$$

While for categorical values the attacker looks for unique values in a column, as:

$$\{a_{\text{categorical}} == v\} \qquad \text{if } \text{len}(\mathcal{D}[a_{\text{categorical}}] == v) == 1$$

This strategy leverages the fact that synthetic data generators may inadvertently replicate rare and unique individuals (outliers) from the training set. These individual attribute predicates are then combined to form a multivariate predicate, which is tested against both the training and control datasets to calculate the privacy risk R.

Linkability

Linkability attacks refer to the category of attacks that, given two sets of disjoint original attributes, try to link them using the synthetic dataset. This attack models a realistic scenario where an attacker possesses one set of publicly available, quasi-identifying data (e.g., demographics) and wishes to link it to a separate, sensitive dataset (e.g., health information) to re-identify individuals and access their private information. To simulate this, the target dataset is split in 2, that is, $\mathcal{D}_1 = \mathcal{D}_{\text{target}}[a_1, a_2, \ldots, a_k]$ and $\mathcal{D}_2 = \mathcal{D}_{\text{target}}[a_{k+1}, a_{k+2}, \ldots, a_N]$*. The attack then proceeds on a per-record basis. For each

*Although the indices are written in order for clarity, the attributes can be divided in any arbitrary manner. So, the subsets do not need to follow sequential ordering.

row of the splits, its nearest neighbor in the synthetic dataset is computed with respect to the Gower distance [54]:

$$D_{\text{Gower}}(d_i, d_j) = \frac{1}{N}\left(\sum_{\text{numeric}} \frac{|d_{ik} - d_{jk}|}{Range(k)} + \sum_{\text{categorical}} s(d_{ik}, d_{jk})\right) \tag{3.4}$$

where d_i and d_j are two arbitrary records, d_{ik} is the value of the k-th column of d_i, and the similarity measure for categorical attributes $s(\cdot, \cdot)$ is defined as:

$$s(d_{ik}, d_{jk}) = \begin{cases} 0, & \text{if } d_{ik} = d_{jk} \\ 1, & \text{otherwise} \end{cases} \tag{3.5}$$

The Gower distance allows one to compute a distance that takes into consideration the mixed data types that are common in structured datasets. From the nearest neighbors, 2 sets of $n_{\text{neighbors}}$ indices are obtained, one from each split, which are denoted by $I_{\mathcal{D}_1}$ and $I_{\mathcal{D}_2}$. The attack is successful if the intersection between the two sets of indices is not the null set. From a practical point of view, if this attack is successful, then an attacker is able to link two different sources of information through synthetic data.

Inference

The Inference attack assumes the attacker possesses external knowledge, termed *auxiliary information*, which consists of a subset of known attribute values for a target individual (e.g., from a separate, public dataset). The attacker's goal is to leverage this partial information, in conjunction with the synthetic data, to infer the unknown value of one or more sensitive target attributes. For example, an attacker might know an individual's occupation, education level, age, and sex (the auxiliary information) and, with the help of the synthetic dataset, attempt to infer their net income or location (the sensitive target attributes).
To execute this, the attacker, using the auxiliary information, will search for the nearest neighbor that possesses (or almost possesses) the specific combination of values for the known attributes to infer the values of the target attributes. Again, the Gower distance is employed to compute the distances between records.
The attack is successful if, for a categorical attribute, the inferred value matches exactly the true value in the original dataset. For numeric attributes, the inference is considered successful if the inferred value is within a pre-defined tolerance δ of the true value (e.g., $|v_{\text{inferred}} - v_{\text{true}}| \leq \delta$), where δ is chosen based on the attribute's scale.

3.3 Contrastive Learning-based solution

The approach proposed in [99] suggests using AI to improve the creation of these attacks. In particular, a contrastive learning-based approach is employed.

The central hypothesis of this work is that the original Anonymeter Singling Out attack is overly simplistic. It relies on hand-crafted heuristics (e.g., checking min/max values) that only identify obvious, one-dimensional outliers. We argue that true vulnerability lies in complex, multi-attribute correlations that make a record unique.

Our proposed approach is designed to identify these complex outliers. By running a local outlier detector (LOF) in the learned contrastive embedding space, we are no longer searching for simple min/max values. Instead, we are finding records that are semantically isolated in a space that understands the data's correlations. These records are much more likely to be true "memorized" individuals from low-density regions of the training data.
We extend our contrastive learning approach to also address Linkability and Inference attacks, testing these extensions exclusively on the Adult dataset. This evaluation revealed that the approach requires significant computational resources to be practically applied to these advanced attack vectors, indicating a limitation for resource-constrained environments.

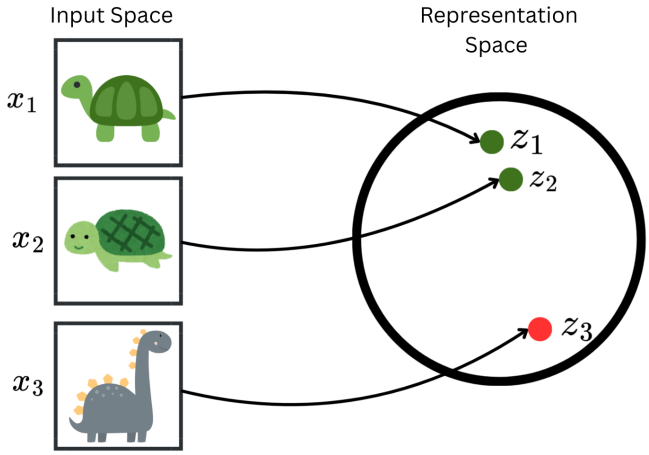


Figure 3.2: Visualization of contrastive learning embedding generation. The model maps similar inputs to proximal points in the representation space, while pushing dissimilar inputs apart.

Contrastive Learning

Contrastive learning is a ML technique that focuses on creating lower-dimensional representations of the data in a self-supervised manner without using labels. As shown in Figure 3.2, the contrastive learning approach learns the representation of the input data such that if two input points are close to each other in the input space, they should also be closer in the learned representation space (positive pairs). Conversely, if two input points are different, then they will be represented far from one another (negative pairs). The model learns by defining these positive and negative samples for each data point through a "self-labeling" process, then optimizing a loss function to maximize the distance between negative pairs and minimize the distance between positive pairs.
This type of approach has been studied in many domains, such as computer vision [26] or natural language processing [133]. However, applying it to tabular data is challenging because, unlike images, tabular data lacks natural augmentation-invariance (such as cropping, rotating). Therefore, a key challenge is defining a meaningful way to create positive pairs from a single row. One such approach for tabular data was proposed by

Shenkar et al. [120]. In this thesis, from each row d_i a set of m pairs is constructed by extracting the first k consecutive attributes in one set and the remaining $n - k$ in the other. For example, the row d_i is divided into the tuple (a_i, b_i), where a_i identifies the first k attributes, while b_i identifies the remaining $n - k$. Then, two maps are learned to maximize the mutual information between the subrows that belong to the same rows, while minimizing it if this is not the case.
The approach we propose in [99] is inspired by this but uses a different augmentation strategy. Let x and y be two distinct records, where $x = (x_1, x_2, \ldots, x_M)$. Let's define x' the masking of x as:

$$x'_i = \begin{cases} x_i, & \text{if column } i \text{ is not masked} \\ \emptyset, & \text{otherwise} \end{cases}$$

To learn representations, we first create two distinct, randomly masked versions for any two given records, x and y, resulting in (x', x'') for x and (y', y') for y. Pairs derived from the same original record, such as (x', x''), are treated as positive pairs. Pairs from different records, such as (x', y''), are treated as negative pairs. We then train a neural network f to map these masked inputs into an embedding space. The network is optimized to make the embeddings of positive pairs "close" while pushing the embeddings of negative pairs "distant," based on a similarity metric. The resultant embedding $f(x')$ is normalized, which is denoted with $f^N(x')$, as shown in Figure 3.3. The masking function is stochastic: it first randomly selects the number of columns to mask (from 1 to $M - 1$), and then randomly selects which columns to mask.

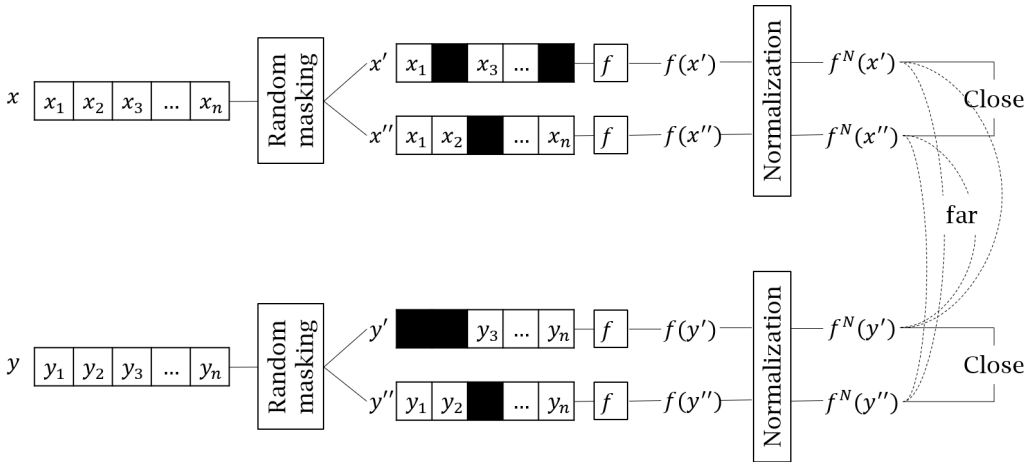


Figure 3.3: Visualization of how the embeddings are created by using the contrastive learning method proposed in [99].

The network architecture begins with an embedding layer for categorical variables, which maps them into a dense numerical space. This is followed by the core network, a 3-layer Feedforward Neural Network with 1024 neurons per hidden layer. To alleviate overfitting, we apply a dropout layer ($p = 0.1$) and a normalization layer. We use the GELU activation function [77], defined as

$$\text{GELU}(x) = x \cdot \Phi(\alpha x)$$

where $\alpha > 0$ is a smoothing factor and is generally set to be $\alpha = 1$, and Φ is the Gaussian cumulative distribution function (CDF). We chose this activation function over the more common ReLU because it avoids critical issues, such as dead neurons and sparse gradients. Since ReLU sets all negative activations to 0 (effectively killing a neuron's signal), its use can be problematic; our chosen function's smoothness, by contrast, leads to more stable training. The similarity metric used in the normalized embedding space is the cosine similarity, defined as:

$$S_c(x, y) = \frac{x \cdot y}{||x||\ ||y||} \tag{3.6}$$

which measures the similarity of two vectors, or embeddings, as the angle between them, without taking into consideration the magnitude of the vectors.

With contrastive learning, as stated in [134], the learning is performed to align positive pairs and disalign negative ones. In this way, the distance between the embeddings is no longer important; rather, their orientation that is, the angle in between is significant. This justification is convenient with the cosine similarity. In fact, the way we are computing it through normalization makes the embeddings lie on the n-dimensional unit hypersphere. In this setting, the cosine similarity (which measures the angle) becomes a natural metric, as it is directly equivalent (up to a constant) to the Euclidean distance.

3.3.1 Singling Out

To improve the original Singling Out attack [49], we propose a method that combines both contrastive learning and the identification of vulnerable records; that is, the individuals or records that are more susceptible to identification risk. First, we use the contrastive learning model (detailed in the previous section) to compute embeddings for all records in the synthetic dataset. Second, we apply the Local Outlier Factor (LOF) algorithm [15], via the Scikit-Learn package [103], to this embedding space. Any synthetic record identified as an outlier by LOF is defined as a "vulnerable record". The LOF algorithm was specifically chosen because it is a density-based method. It identifies outliers based on their isolation relative to their local neighborhood. This is ideal for our purposes, as we are not looking for global anomalies, but for records that are isolated from their own conceptual clusters. These "local" outliers are precisely the ones most likely to represent memorized records. Third, we retrieve these vulnerable synthetic records and use their attribute values to construct attack predicates. For categorical attributes, the predicate is created using the record's exact value (for example, occupation == 'Professor'). For numerical attributes, we first discretize the attribute's full range into bins; the attacker then uses the specific interval in which the outlier's value resides (e.g., age $\in [80, 90]$) as a univariate predicate. These single-attribute predicates are concatenated to form a multivariate predicate. Finally, following the Anonymeter methodology, these generated predicates are tested; only those that successfully single out a unique individual in the training set are kept. These "successful" predicates are then evaluated against both the training and control datasets to compute their respective success rates (r_{train}, r_{control}) and the final normalized risk score R.

3.3.2 Linkability

The Linkability attack models an adversarial scenario in which a set of attributes for a specific individual is divided into two disjoint subsets, and an attacker attempts to re-associate these subsets by utilizing the synthetic dataset as a structural bridge. Unlike traditional linkage methods that rely on direct distance metrics between raw features—which often degrade in high-dimensional or mixed-type settings—our framework leverages the semantic consistency of the learned latent space to identify records that belong to the same identity.

The evaluation process begins with a stochastic partitioning of the attribute space. For a given record, the features are divided into two disjoint sets, effectively simulating an adversary who possesses partial information from two distinct sources. To maintain compatibility with the contrastive learning architecture 3.3, which is optimized for high-dimensional full-record representations, we do not physically truncate the datasets. Instead, we utilize a masking strategy where the irrelevant attributes for a given split are masked, allowing the model to project the partial information into the latent space while preserving the structural integrity required by the encoder. This results in four distinct sets of embeddings: two derived from the training data (representing the partial records to be linked) and two derived from the synthetic data (representing the reference manifold). To implement the search, we treat the synthetic data as a global coordinate system. We construct two high-dimensional search trees using the embeddings of the synthetic dataset corresponding to each split. In our implementation, we employ the BallTree algorithm for its efficiency in high-dimensional metric spaces. These trees represent the distribution of the synthetic population through the lens of the two different attribute subsets. The final linkage determination is based on a topological query. Each training embedding from the first split is used to query its corresponding synthetic search tree, and the process is repeated for the second split. A successful linkage is recorded if both queries converge on the same synthetic record index. This convergence implies that the synthetic record serves as a unique "pivot" point that statistically reconciles the two disjoint pieces of information. By aggregating these successes across various attribute partitions, we can quantify the Linkability risk as the probability that an adversary can uniquely re-identify an original individual by navigating the synthetic latent manifold. To conclude, we quantify this risk using the Wilson Score Interval, ensuring that our final Linkability score reflects a statistically robust lower bound on re-identification probability, accounting for sample variance rather than relying on a potentially unstable point estimate.

3.3.3 Inference

The attribute inference attack simulates a scenario where an adversary possesses partial knowledge of a record and attempts to reconstruct a sensitive target attribute. Our approach formalizes this threat through a multi-stage framework that combines topological retrieval with parametric modeling, using the contrastive latent space to adjudicate between competing predictions.

The process begins with a topological search operation. We first generate embeddings for the original training records (the queries) where the target sensitive attribute is masked to simulate incomplete knowledge. Simultaneously, we project the synthetic

dataset into the same latent space and construct a search tree (utilizing the BallTree algorithm) on these synthetic embeddings. By querying this tree with the masked training embeddings, we identify the synthetic record that acts as the closest semantic proxy. This proxy provides our first inference candidate: a non-parametric retrieval-based prediction, where the target value is imputed directly from the nearest synthetic neighbor. In the second stage, we generate a competing candidate using a parametric inference model. An XGBoost regressor or classifier is trained exclusively on the synthetic dataset to learn the functional mapping between known features and the target attribute. This model is then applied to the features of the synthetic neighbor to produce a model-based prediction.
To resolve the potential conflict between these two candidates, we implement a contrastive ranking mechanism based on latent similarity. We represent each candidate prediction by creating a "padded record" that contains only the predicted target attribute value, which is then projected into the embedding space. To evaluate which candidate is more semantically consistent with the original query, we compute the logits via a dot product (implemented using the einsum operation) between the embedding of the partial training record and the embeddings of the two candidate predictions. The candidate that maximizes this similarity score is selected as the final inference. This dual-pathway approach ensures that the final prediction is not only statistically probable according to the synthetic distribution but also maintains the highest degree of semantic alignment with the specific context of the individual record.

3.3.4 Distance to Closest Record

In addition to attack-based metrics, this chapter also evaluates a prominent similarity-based metric, namely the Distance to Closest Record (DCR). Similarity-based metrics assess privacy by quantifying the resemblance between the real (training) and synthetic datasets. The underlying assumption is that synthetic records found to be too similar to real records can lead to identity disclosure or the inference of sensitive attributes [28, 49, 51].
The DCR is a function that takes two datasets as input: a real dataset $\mathcal{D}$ and a synthetic dataset $\hat{\mathcal{D}}$, and returns a real-valued score representing the level of similarity between them:

$$\text{DCR} : \mathcal{D} \times \hat{\mathcal{D}} \to \mathbb{R}.$$

The computation of DCR relies on comparing two distance distributions: the Synthetic-to-Real Distance (SRD) and the Real-to-Real Distance (RRD). Given a real (training) dataset $\mathcal{D}$ and a synthetic dataset $\hat{\mathcal{D}}$, the SRD is defined as:

$$\text{SRD}(\hat{d}) = \min_{d \in \mathcal{D}} Dist(\hat{d}, d) \tag{3.7}$$

Thus, for each record in the synthetic dataset, we find the distance to its nearest neighbor in the real training data. In the computation of the RRD instead, a holdout set of the real data is required. Let's denote this holdout set as $\mathcal{D}_2$ and the remaining records as $\mathcal{D}_1$, so that $\mathcal{D}_1 \cap \mathcal{D}_2 = \emptyset$. The RRD is then defined as the distance from each record in $\mathcal{D}_1$ to its nearest neighbor in $\mathcal{D}_2$:

$$\text{RRD}(d_1) = \min_{d_2 \in \mathcal{D}_2} Dist(d_1, d_2) \tag{3.8}$$

The core privacy test involves comparing these two distributions. Suppose now that, for a synthetic record $\hat{d}$, we have:

$$\text{SRD}(\hat{d}) < \text{RRD}(d^*), \text{ for } d^* = \arg\min_{d \in \mathcal{D}} \text{Dist}(\hat{d}, d). \tag{3.9}$$

This inequality implies that the record $\hat{d}$ is closer to its nearest neighbor in the training set than that training set record is to any other record in the holdout set. This suggests that $\hat{d}$ may leak information because of its high similarity to the real data.
DCR is derived from a statistical comparison between the SRD and RRD distributions. In the majority of the works that use or cite DCR (e.g. [13, 101, 105]), the DCR is computed through the comparison of quantiles between the two distributions. In this way, DCR provides a measure of what percentage of SRD values are smaller than the smallest percentile rank of RRD values. In our experiments, we preprocess our data to allow using Euclidean distance as the similarity metric. In particular, numerical variables are rescaled using the Standard Scaler, while for categorical variables, Ordinal Encoder or Label Encoder are used, all of them from Scikit-Learn [103]. Finally, we consider two distinct datasets, namely $\mathcal{D}$ and $\hat{\mathcal{D}}$, the DCR metric is computed as:

$$\text{DCR}(\hat{\mathcal{D}}, \mathcal{D}) := \frac{\left|\left\{\hat{d} \in \hat{\mathcal{D}} : \text{SRD}(\hat{d}) < \text{RRD}_\alpha\right\}\right|}{\frac{\alpha}{100} \cdot |\mathcal{D}_1|} \tag{3.10}$$

where α is a percentage (we set it to 2% but in other works there are also used 1% and 5%), and RRD_α is the α^{th} percentile of the RRD distribution. In particular, in the numerator, we are computing how many records from the synthetic dataset have a smaller SRD than RRD_α. This value is then normalized using the number of real records that have an RRD below the defined threshold.
To create a final, interpretable score, this DCR value is normalized using Equation 3.11 (from [105]). This "Privacy Score" is designed to be bounded between 0 (indicating no privacy risk) and 1 (indicating maximum risk). A score of 1 represents a worst-case scenario where all synthetic records are found to be closer to the training data than the α-percentile baseline. Conversely, a score of 0 is the ideal outcome, representing the case where the synthetic data is statistically independent of the training set (i.e., no memorization is detected).

$$\text{Privacy Score}(\hat{\mathcal{D}}, \mathcal{D}) = \frac{\frac{\alpha}{100}(DCR(\hat{\mathcal{D}}, \mathcal{D}) - 1)}{1 - \frac{\alpha}{100}} \tag{3.11}$$

However, it is important to note that since this score is a statistical estimator, it is subject to sampling variance. In a true "no risk" scenario (where the synthetic data is independent), the estimator has an expected value of 0, but a non-zero variance. Therefore, due to random sampling effects, the measured score can occasionally be a small negative value. This is not an error, but a normal statistical artifact. Any score at or slightly below 0 should be interpreted as "no detectable privacy risk", meaning the metric cannot distinguish the synthetic data from an independent, non-leaky dataset.

Summary of Theoretical Claims and Experimental Validation.

The framework proposed in Sections 3.3.1 to 3.3.4 rests on three central theoretical claims that distinguish it from the state of the art. First, we posit that privacy vulnerabilities in deep generative models manifest as semantic outliers—records isolated in the latent manifold—which cannot be detected by the simple heuristic predicates used in standard Singling Out attacks. Second, we hypothesize that mapping records into a contrastive embedding space captures non-linear attribute correlations, thereby enabling more sensitive Linkability and Inference attacks than those based on rigid distance metrics (e.g., Gower distance). Third, we anticipate a divergence between distance-based and density-based metrics in the embedding space, specifically that the contrastive loss optimizes for angular alignment rather than Euclidean proximity.
In the subsequent experimental evaluation, we empirically validate these claims by benchmarking the sensitivity of the Contrastive Learning framework against the Anonymeter baseline. We specifically test the framework's ability to detect increasing levels of memorization in controlled "overfitting" scenarios and quantify the trade-off between the achieved statistical gain and the computational overhead required to train the embedding networks.

3.3.5 Experimental Evaluation and Datasets

To validate the robustness of the proposed contrastive privacy metrics, we conduct a series of experiments across diverse tabular environments. This evaluation transitions from highly controlled synthetic scenarios to real-world generative modeling, allowing us to benchmark our framework against established baselines.

Datasets and Preprocessing

We use three primary datasets commonly used in the privacy-preserving literature: Adult [7], Texas Inpatient Public Use Data ("Texas") [21], and the 1940 Census full enumeration ("Census") [1]. The Adult dataset consists of 48,842 records and 15 attributes, including categorical variables such as race and occupation, and numerical features like *age* and *capital-gain*. Following the methodology of the Anonymeter framework [49], we utilize specific attribute subsets for the larger Texas and Census datasets (the same 28 and 37 attributes, respectively). However, to accommodate the memory and computational requirements of our contrastive models while maintaining statistical significance, we resized the Texas and Census datasets to 60.000 and 75.000 records, respectively. To ensure a fair comparison, we re-ran the standard Anonymeter attacks on these modified versions to establish an updated baseline.

Controlled Scenarios: Leaky and Noisy Synthesizers

The first stage of our evaluation employs controlled scenarios where the degree of ground-truth privacy leakage is known. In the **Leaky Synthesizer** setup (propose in [49]), the original data is partitioned into three non-overlapping sets: training, control, and release. A synthetic dataset is then constructed by mixing training and release records according to a predefined leak fraction (l_f).

To better approximate the behavior of real-world generators, which rarely replicate training records exactly, we introduce a novel **Noisy Synthesizer**. In this configuration, we apply a noise-injection function $\psi(\cdot, \sigma, \lambda, p)$ to the training records before they are integrated into the synthetic set. We define three distinct noise perturbations to handle the varied nature of tabular data:

- Real noise parameter (σ): This parameter is used on numeric attributes that are real numbers. In this case, given the original value v, we create the new value by adding Gaussian noise to it. The new value is $v_{\text{noisy}} = v + \mathcal{N}(0, \sigma)$.
- Integer noise parameter (λ): This parameter is used for integer numeric values. We distinguish between integer and real because of their different domains and because the order still matters (which may not be the case with unordered categorical variables). The new value is computed by adding, or removing an integer number sampled from a Poisson distribution with λ as its parameter. So, $v_{\text{new}} = v + s \cdot Poisson(\lambda)$, where s is a random variable with a Rademacher distribution[†].
- Categorical noise parameter (p): For categorical attributes, p denotes the probability of switching the original value to another value chosen uniformly at random from the set of all possible categories for that attribute.

We test all privacy metrics against the Noisy Synthesizer by systematically evaluating the effect of noise injection across all 2^3 configurations of the parameters (σ, λ, p), utilizing 0.05 as the non-zero value for parameter perturbation. This includes the two defining cases: the non-noisy baseline ($\sigma = \lambda = p = 0.0$) and the maximal perturbation case ($\sigma = \lambda = p = 0.05$). This value of 0.05 was chosen empirically; preliminary tests showed that 0.01 was too low to have a meaningful effect, while 0.1 corrupted the data too much. Formally, the synthetic dataset is now created in the following way:

$$\mathcal{D}_{\text{synth}} = \text{concat}\left[\psi(\mathcal{D}_{\text{train}}^{l_f}; \sigma, \lambda, p), \mathcal{D}_{\text{release}}^{1-l_f}\right] \tag{3.12}$$

where "concat[...]" is the concatenation function that concatenates the dataframes, $\mathcal{D}_{\text{train}}^{l_f}$ is the portion of training copies used in the synthetic dataset, $\mathcal{D}_{\text{release}}^{1-l_f}$ is the portion of records that came from the release set, and $\psi(\cdot; \sigma, \lambda, p)$ is the noise-injection function previously defined.

Generative Modeling and Overfitting Analysis

For the deep learning evaluation, we again test the attack on the synthesizers used in the Anonymeter paper [49], so CTGAN [137] and DPCTGAN [40]. We extend this evaluation in two significant ways. First, we add another synthesizer that, in our test, has a better performance in generation, which is REaLTabFormer [122]. The second addition regards this new added model. We want to test how all the metrics behave if the synthetic data generator is trained at different levels of overfit, which implies a different amount of copies (or almost copies) of training records in our generated dataset. To better explain

[†]The Rademacher distribution is a discrete probability distribution where a random variable can assume the values in {-1, +1} with a probability of 50% each.

this, we introduce the *overfit ratio*, a value that represents how much we are overfitting a synthetic data generator. The overfit ratio is obtained using the validation loss computed during model training. So, it is defined as:

$$\text{Overfit Ratio}(e) = \frac{\mathcal{L}_{\text{val}}(e)}{\min \mathcal{L}_{\text{val}}} \tag{3.13}$$

where $\mathcal{L}_{\text{val}}$ denotes the validation loss score, e indicates the epoch in which we want to compute the overfit ratio, and $\mathcal{L}_{\text{val}}(e)$ is the value of the validation loss at a specific epoch. This measure is used to account for the overfitting of a model, taking into consideration the behavior of the validation loss of a generic AI model. In fact, during the training phase, the validation loss will decrease in the initial phase of the training; then, normally, when it reaches a minimum, the early stopping is triggered and the training procedure is stopped. The early stopping verifies if the validation loss has reached a minimum, continuing the training a couple of epochs after the "hypothetical" minima is found, to make sure that, after it, the validation loss starts to increase or not. If the increase continues for more than a predefined number of epochs (called patience), the training is stopped, and the model's parameters relating to the model for which we have the minimum validation loss are saved. Knowing that overfitting is one of the main causes of privacy leaks when generating data with deep learning methods, we decide also to test the metric in this environment in which we deliberately overfit the model to see how the privacy metric behaves and if their trend matches our expectations. In this setting, we train RTF at 6 different levels of overfit, that is, 1.0, 1.2, 1.4, 1.6, 1.8, and 2.0. We expect that all privacy metrics have an increasing trend along with the overfitting ratio [19, 139].
We will refer to this experiment setup as the overfitting-scenario or overfitting-experiment and want also to make clear that CTGAN has not been used in this setting because the implementation and the model's type did not allow for a straightforward and meaningful computation of the validation loss and the overfit ratio.
In summary, the experimental setup counts on 3 distinct models, namely CTGAN, DPCTGAN, and RTF, from which only one uses the differential privacy mechanism during its training. Regarding the experiments, we run the noisy synthesizer, the leaky synthesizer (which can be seen as a specific case of the noise synthesizer with all noise parameters set to 0), the straightforward synthetic data generation with 3 models, and, to conclude, the overfitting experiment using only RTF.

3.4 Results

This section is organized into two distinct components to provide a granular assessment of privacy risks. The first subsection presents the Singling Out evaluation, adhering to the comprehensive protocol established in the primary study. The subsequent subsection extends our analysis to Attribute Inference and Linkability attacks; notably, for these latter assessments, we restrict our experimental scope exclusively to the Adult dataset to allow for a focused investigation into these complex leakage modalities.

To generate the embeddings required for these evaluations, we trained our proposed contrastive learning network for 300 epochs using early stopping regularization, imple-

mented with a window of 20 epochs and a patience of 10. The learning rate was fixed at 10^{-3}. Furthermore, to accommodate varying feature complexities across the datasets used in the Singling Out, we scaled the embedding dimension m accordingly: setting it to 10 for the Adult dataset, 20 for Texas, and 30 for Census.
Regarding the generative models, for CTGAN and DPCTGAN we use the same hyperparameters used in [49]. For RTF, default parameters are used with the addition of setting the *trainsize* parameter to 0.9 to let the model have also a small portion of data to use in the validation phase at the end of each epoch. For the overfitting experiment, we turn off all the regularization, namely the dropout (the *resid pdrop* for the fully connected layers, the *embd pdrop* for the dropout ratio for the embeddings, and the *attn pdrop* for the attention layer). In addition, the *greater is better* parameter is set to *True* in order to save the parameters of the model with the highest validation loss; this is done because normally the saved model corresponds to the one with the lowest validation loss. Finally, the early stopping is turned off in order to let the model continue the training after the minimum point of the validation loss is found.

3.4.1 Singling Out and DCR

For the Singling Out (SO) attack, we use Anonymeter's default parameters. We generate 2000 attacks and ran both the univariate and the multivariate attack with the number of variables that range between 3 and 12. This is essentially done because of the highly demanding computation of the attack that, in order to create the predicates, involves inspecting all rows and combinations of unique attribute-value pairs. We report the risk along with their confidence interval (CI) calculated as highlighted in Section 3.2. For DCR, we compute the confidence interval using bootstrapping repeating the measurements $n = 1000$ times. Computing the confidence interval in this way usually yields a small value, that is, a value around 10^{-3} or even 10^{-4} which makes the CI not always visible in the plots.
We also introduce another metric in the evaluation which we will denote as "DCR + CL", where we compute the distance to the closest record as explained in Section 3.3.4, but perform the nearest-neighbor search within the embedding space computed by our contrastive learning method, rather than using the Euclidean distance on the preprocessed data. This is done to test if our learned representation can also improve the performance of similarity-based metrics.

Leaky and Noisy Synthesizer

We start the discussion with the noisy synthesizer. The results are shown in Figure 3.4. The x-axis represents the leak fraction, while the y-axis reports the risk magnitude. Then, for each row, we set a different configuration of noise injection, which is denoted by a tuple of 3 numbers that represents respectively σ, λ and p. In the first row, all the noise parameters are set to 0, which corresponds to the original leaky synthesizer. While in the leaky setup we don't see any difference between the metrics, the noise injection seems to make it more difficult to identify a privacy leak. In fact, adding noise only to the numeric variables makes the original Singling Out struggle with respect to the other methods. By adding noise to integer values instead, leads to a decrease of the risk of the

similarity methods, while the attack ones have not a significant difference between each other. At last, the probability of switching categories seems to not affect any metric.
From this experiment, we can clearly see that all the methods have the desired trend, that is, the risk is increasing along with the leak fraction. We can also see that our method is constantly outperforming the state of the art method, detecting a higher risk measure even if the noise seems more impactful as in the case of injecting noise to numeric variables.

Real Generative Model Comparison

The results are highlighted in Table 3.1. While elapsed times are shown in Table 3.2. These results present a more complex picture. For the RTF model on the two larger datasets (Texas and Census), our proposed CL + SO attack successfully identifies a higher privacy risk than the baseline SO attack. However, for all CTGAN and DPCTGAN models, and for RTF on the Adult dataset, our CL + SO attack reports a lower risk than the baseline.
The DCR + CL metric consistently reports a risk score near zero, similar to or slightly lower than the baseline DCR, suggesting that the learned embedding space does not provide an advantage for this type of similarity metric.

Dataset	Method	SO [49]	CL + SO	DCR	CL + DCR
Adult	CTGAN	0.12431 ± 0.018	0.08193 ± 0.024	$-0.00843 \pm 3.9 \times 10^{-4}$	$-0.01048 \pm 4.4 \times 10^{-4}$
	DPCTGAN	0.11184 ± 0.017	0.09916 ± 0.015	$-0.01993 \pm 2.2 \times 10^{-5}$	$-0.01996 \pm 2.3 \times 10^{-5}$
	REalTabFormer	0.02783 ± 0.031	0.01984 ± 0.028	0.00467 ± 0.001	$-0.01815 \pm 1.1 \times 10^{-4}$
Texas	CTGAN	0.01537 ± 0.015	0.01115 ± 0.009	$-0.02020 \pm 2.6 \times 10^{-5}$	$-0.01061 \pm 6.0 \times 10^{-4}$
	DPCTGAN	0.00816 ± 0.009	0.00611 ± 0.009	-0.02040 ± 0.0	$-0.01245 \pm 2.7 \times 10^{-4}$
	REalTabFormer	0.03103 ± 0.026	0.04340 ± 0.029	$-0.01893 \pm 9.6 \times 10^{-5}$	$-0.01194 \pm 1.9 \times 10^{-4}$
Census	CTGAN	0.01340 ± 0.021	0.00783 ± 0.005	$-0.01961 \pm 4.3 \times 10^{-5}$	$-0.01943 \pm 1.0 \times 10^{-4}$
	DPCTGAN	0.01077 ± 0.009	0.00871 ± 0.005	-0.02040 ± 0.0	$-0.01999 \pm 3.6 \times 10^{-5}$
	REalTabFormer	0.02695 ± 0.024	0.04650 ± 0.027	$0.01882 \pm 4.88 \times 10^{-4}$	$0.01008 \pm 5.1 \times 10^{-4}$

Table 3.1: Measured leakage risk using various metrics: SO (Singling Out Attack), DCR (Distance to Closest Record), and CL (Contrastive Learning embeddings). The results indicate that for DPCTGAN-generated synthetic data on the Texas and Census datasets, the DCR metric yields a uniformly low risk (no record below RRD_α). This uniform result eliminates risk measure variability in the bootstrapping procedure.

Finally, we analyze the computational efficiency of the proposed methods, as detailed in Table 3.2. The results highlight a significant trade-off between the two approaches. For the Singling Out (SO) attack, our proposed CL + SO method demonstrates a notable efficiency gain on two of the three datasets. On the Adult dataset, our method reduces the execution time from 2842 seconds to 863 seconds (a $3.3\times$ speedup), and on the Census dataset, it reduces it from 2167 seconds to 1645 seconds. This improvement is likely due to the algorithmic difference: the original SO attack relies on a brute-force search for unique attribute combinations, which suffers from combinatorial explosion as the number of attributes increases. In contrast, our method shifts this complexity to the training of the neural network and the subsequent application of the LOF algorithm, which scales more favorably with dataset complexity. The exception is the Texas dataset, where CL + SO was slower (1176s vs. 376s), potentially due to the specific convergence

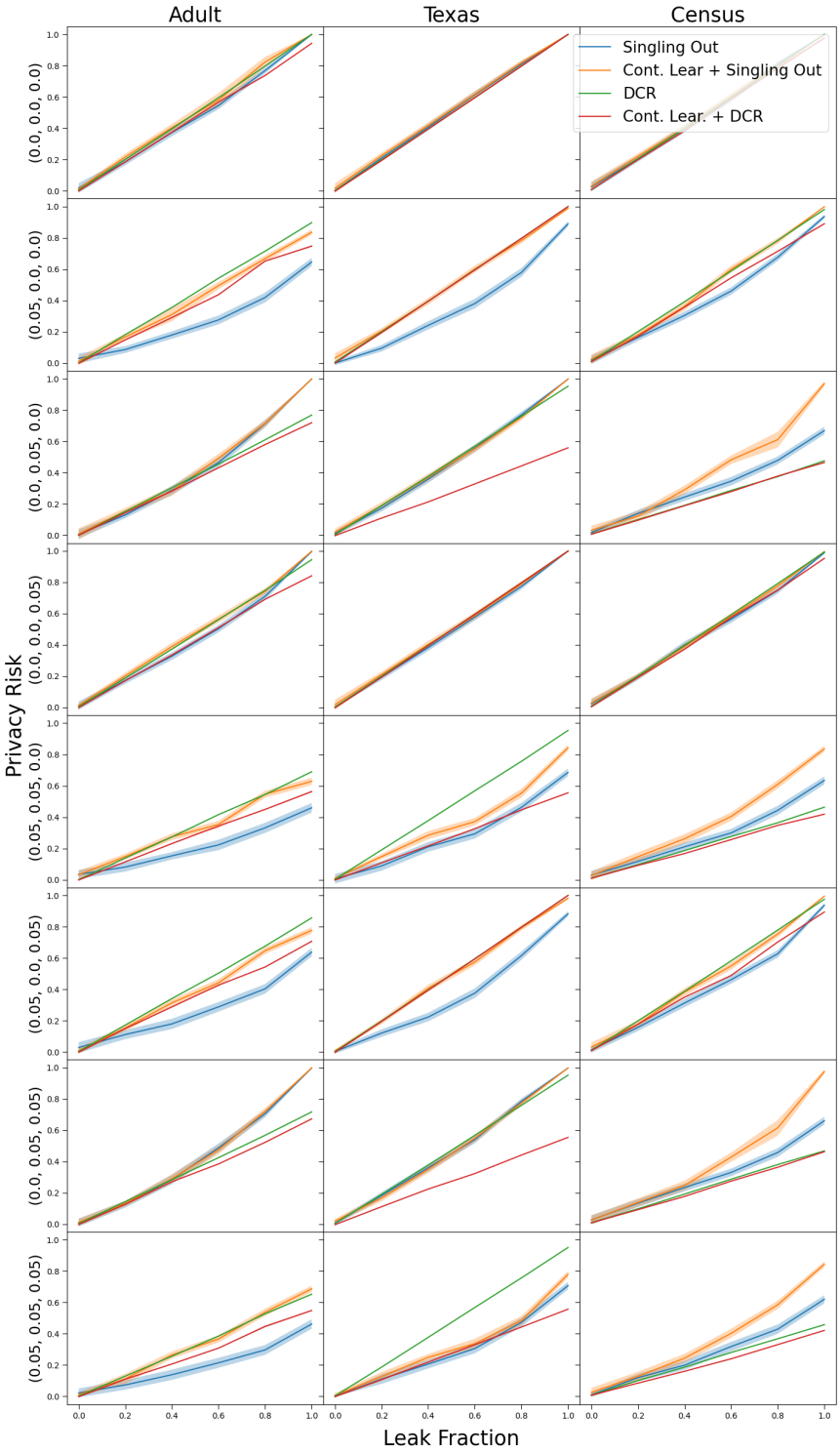


Figure 3.4: Leaky and noisy synthesizer evaluation

behavior of the model or the overhead of the embedding process relative to the ease of finding predicates in that specific dataset.
Conversely, for the Distance to Closest Record (DCR) metric, the addition of contrastive learning (DCR + CL) consistently increases the computational cost across all datasets (e.g., increasing from 7s to 238s on Adult). This is expected, as the baseline DCR is a simple Euclidean distance calculation on the raw data, whereas DCR + CL incurs the significant overhead of training the neural network before any distances can be computed. Given that DCR + CL provided no improvement in privacy risk detection (as shown in Table 3.1), this added computational cost further discourages its use.

Dataset	DCR	CL + DCR	SO [49]	CL + SO	Linkability [49]	Inference [49]
Adult	6.91	238.16	2842.78	863.95	18.06	163.23
Texas	134.72	460.65	376.82	1176.21	103.01	1380.70
Census	83.35	364.40	2167.73	1645.03	99.26	1603.49

Table 3.2: Comparison of the measured time for three methods: SO (Singling Out attack) from [49], DCR (Distance to Closest Record metric), and CL (Contrastive Learning embedding used in conjunction with DCR).

Overfitting Scenario

Finally, the results of the Overfitting Scenario (Figure 3.5) provide compelling validation for our proposed metrics. Across all three datasets, we observe a strong, positive correlation between the Overfit Ratio (x-axis) and the measured privacy risk (y-axis). This confirms our fundamental hypothesis: as the model continues to train beyond the point of optimal generalization (Overfit Ratio > 1.0), it begins to memorize the training data, leading to a monotonic increase in privacy leakage.
The behavior of our proposed Contrastive Learning Singling Out (CL + SO) attack (represented by the purple line) varies by dataset. On the Texas dataset (center plot), CL + SO demonstrates superior sensitivity. As the overfit ratio exceeds 1.6, the CL + SO risk spikes sharply, surpassing the baseline Singling Out attack (blue line) and ending as the highest risk metric at ratio 2.0. This suggests that for this dataset, the embedding space successfully captures the "semantic memorization" that occurs during deep overfitting. However, on the Census dataset (right plot), while CL + SO still trends upward, it consistently reports a lower risk than the baseline Singling Out attack. This discrepancy likely arises from the high dimensionality and sparsity of the Census data, where the baseline's discrete predicate search is more efficient at isolating unique records than the density-based LOF method in the continuous embedding space.
Furthermore, these plots definitively confirm the ineffectiveness of the DCR + CL metric (brown line). On both the Texas and Census datasets, while the standard DCR (red line) correctly identifies increasing risk, the DCR + CL line remains nearly flat and close to zero. This "flatlining" indicates that the contrastive loss, in optimizing the embedding space for semantic similarity (such as for outlier detection), unintentionally decouples the embedding similarity measure from the DCR metric's requirements. This disparity

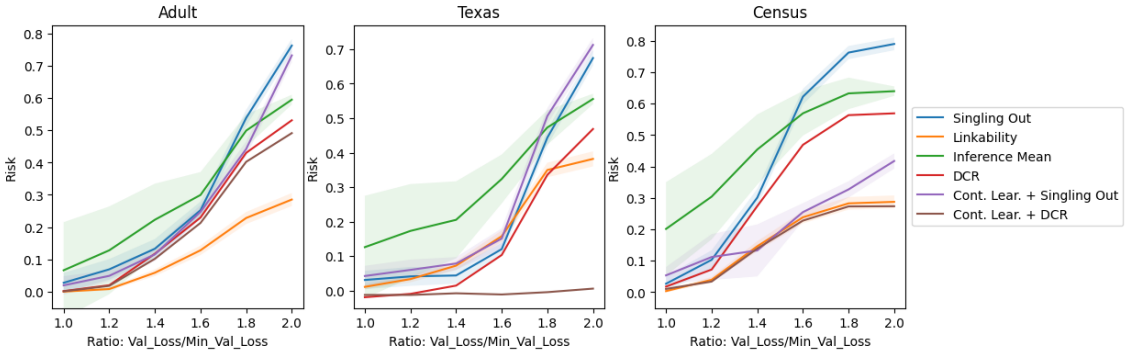


Figure 3.5: Overfit Scenario results

renders the DCR metric insensitive to the increasing data leakage risk it was intended to measure.

3.4.2 Comparative Summary

To address the trade-offs between the proposed Contrastive Learning (CL) framework and the baseline Anonymeter Singling Out, we summarize the key scenarios where our approach offers a distinct advantage.

The most significant practical advantage of the CL framework is its efficiency in the Singling Out attack on large or high-dimensional datasets. As demonstrated in Table 3.2, the CL + SO method achieved a 3.3× speedup on the Adult dataset and a 24% reduction in execution time on Census compared to the baseline. This is because the baseline relies on a combinatorial search for predicates, which scales poorly with data complexity. In contrast, our method shifts the computational burden to the fixed-cost training of a neural network, making it a more scalable solution for auditing large-scale tabular data.

In terms of risk detection, the CL framework demonstrates comparable sensitivity to the baseline in most scenarios, with specific nuances. In the overfitting experiment on the Texas dataset (Figure 3.5), the CL + SO metric effectively tracked the baseline, identifying the same sharp rise in privacy risk as the model memorized the training data. This confirms that the learned embedding space successfully preserves the "semantic memorization" signal required to detect deep leakage. However, on high-dimensional, sparse datasets like Census, the baseline's discrete predicate search remains more effective than the density-based LOF method used in our framework.

The evaluation also highlights clear boundaries for the proposed framework. The CL approach does not provide a benefit for distance-based metrics (DCR) or the Attribute Inference and Linkability attacks, where the high cost of training the network outweighs the statistical parity observed in the results. Therefore, we recommend a hybrid auditing strategy: utilizing the CL + SO framework for rapid and deep Singling Out assessment on large datasets, while retaining the standard Anonymeter protocols for Linkability and Inference tasks.

3.4.3 Linkability and Inference

To validate the efficacy of our proposed framework, we benchmarked our Contrastive-Learning approach against the other two attacks of the current state-of-the-art tool, Anonymeter. We evaluated both methodologies on the Adult dataset using the noisy synthesizer first with the same settings as the previous experiments.

Linkability

As illustrated in Figure 3.6, our method demonstrates a substantial improvement in detecting re-identification risks. Across all evaluated noise configurations, our attack consistently yields higher Linkability scores than the baseline. Notably, in the non-privatized setting $(0.0, 0.0, 0.0)$, our approach uncovers a risk factor nearly 30% higher than Anonymeter. This performance gap highlights the limitations of rigid distance metrics employed by the baseline, which struggle to map disjoint feature splits in high-dimensional spaces. In contrast, our contrastive embedding approach successfully captures the semantic topology of the records, allowing for more accurate linkage even when feature overlap is minimal.
While the CL approach offers a theoretical advancement in capturing semantic leakage, a critical analysis of the experimental results highlights significant practical limitations regarding computational feasibility and statistical gain as shown in Table 3.3. The proposed method incurs a prohibitively high computational cost—approximately 26 times slower than the baseline—rendering it less feasible for rapid auditing. Furthermore, due to overlapping confidence intervals in most scenarios, our approach performs comparably to the state-of-the-art, demonstrating a clear statistical advantage only in the RTF configuration.

Method	Time	CTGAN	DPCTGAN	RTF
Anonymeter [49]	18.06	0.0009 ± 0.0024	0.0009 ± 0.0019	0.0005 ± 0.0022
CL approach	468.06	0.0008 ± 0.0012	0.0003 ± 0.0003	0.0018 ± 0.0014

Table 3.3: Linkability time and results using the CL approach

Inference

The results for attribute inference (Figure 3.7) further corroborate the robustness of our hybrid mechanism. While the baseline performs competitively in low-noise environments, our method consistently establishes a stricter upper bound on privacy risk, tracking equal to or strictly higher than Anonymeter across all trials. The advantage of our approach becomes particularly evident in regimes with higher privacy noise (e.g., configuration $0.05, 0.05, 0.05$), where the dynamic selection between parametric (XGBoost) and non-parametric (nearest neighbor) candidates allows our model to exploit subtle statistical correlations that a purely memory-based attack would miss. Consequently, our framework provides a more rigorous and reliable audit of the synthetic data's vulnerability.

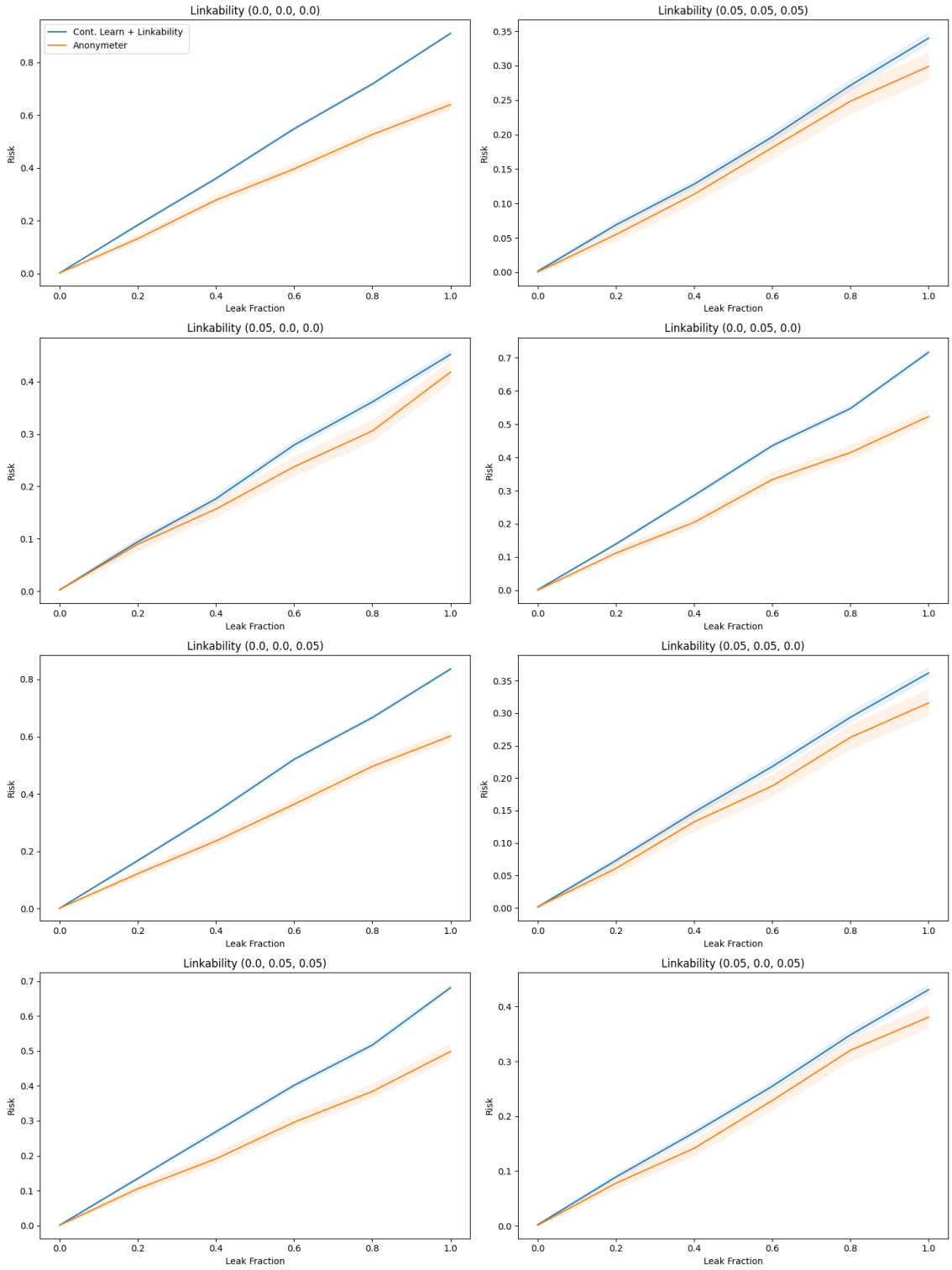


Figure 3.6: Linkability in the leaky and noisy synthesizer experiment

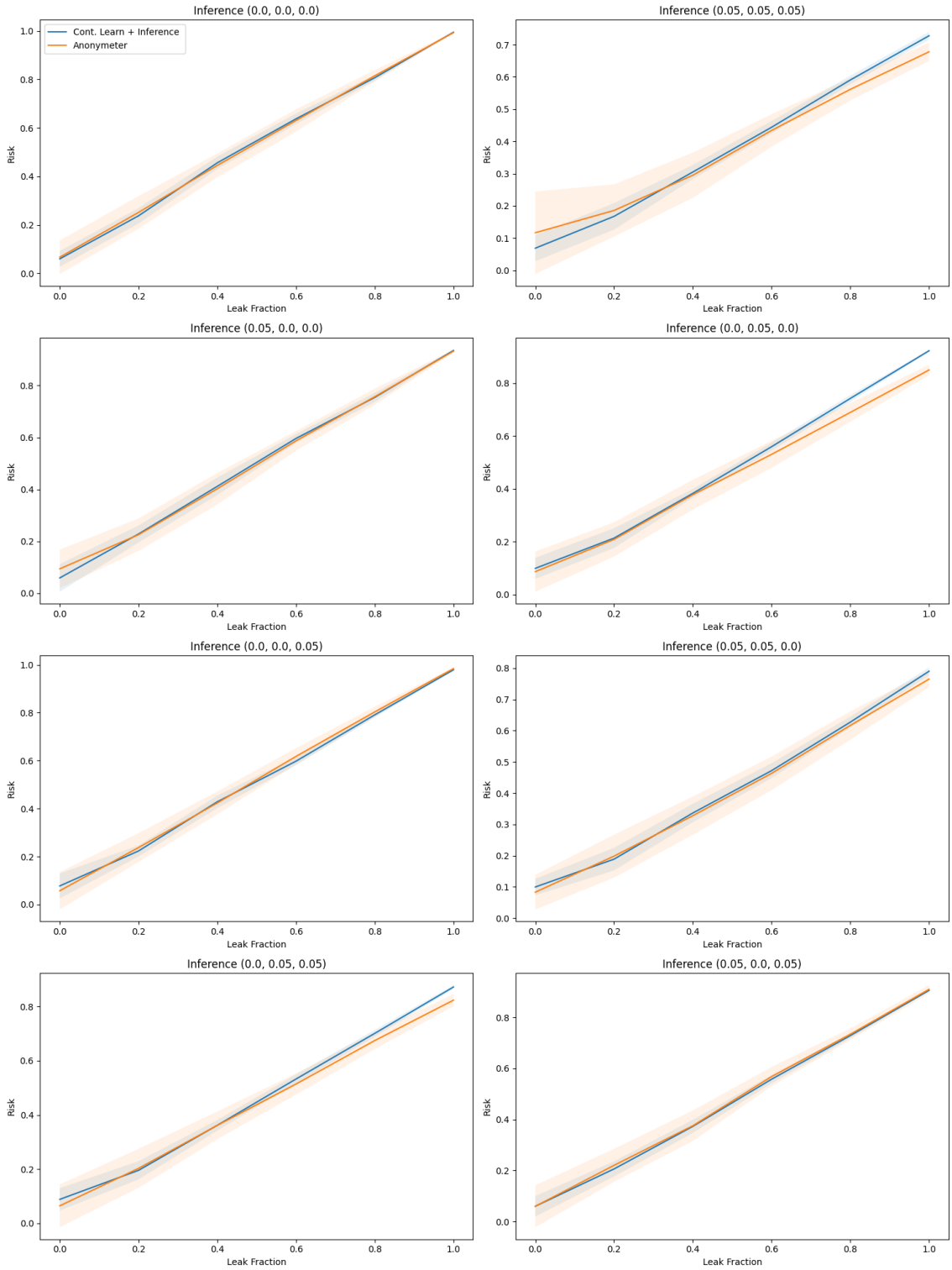


Figure 3.7: Linkability in the leaky and noisy synthesizer experiment

In the context of Attribute Inference, the proposed method is computationally prohibitive, requiring over 50 times the execution duration of the baseline (8642s vs. 162s). Furthermore, the results are statistically indistinguishable across all configurations due to substantial overlap in confidence intervals, indicating that the added complexity yields no significant improvement in risk detection over the state-of-the-art as is presented in Table 3.4.

Method	Time	CTGAN	DPCTGAN	RTF
Anonymeter [49]	162.23	0.0544 ± 0.0622	0.0321 ± 0.0272	0.0973 ± 0.1163
CL approach	8642.81	0.0298 ± 0.0209	0.0081 ± 0.0118	0.0852 ± 0.0407

Table 3.4: Inference time and results using the CL approach

3.5 Conclusion

Our experimental evaluation demonstrates that the proposed Contrastive Learning (CL) framework offers distinct advantages depending on the specific privacy attack being modeled. For Singling Out, the method delivers robust performance improvements without incurring a significant computational overhead, establishing it as a viable and efficient enhancement for standard privacy auditing. Conversely, the application of this framework to Linkability and Attribute Inference reveals critical trade-offs; while the hybrid mechanisms theoretically offer a more granular risk assessment, they do not currently yield statistically significant improvements over state-of-the-art baselines to justify their prohibitive computational costs. Consequently, future research will prioritize optimizing the algorithmic efficiency of these attacks and refining the latent space utilization, with the aim of ensuring that the additional computational investment translates into a tangible and rigorous increase in privacy risk detection.

We further contextualize our findings by benchmarking against the Distance to Closest Record (DCR) metric. Empirically, we observed a strong convergence between the two approaches; across nearly all evaluated scenarios, DCR exhibits behavioral patterns almost identical to our proposed method, reporting statistically equivalent estimates of privacy risk. A distinct trade-off emerges, however, regarding computational efficiency versus interpretability. While DCR benefits from significantly lower latency—inherent to its nature as a direct similarity metric—it suffers from a lack of semantic transparency. The difficulty in translating raw distance values into actionable privacy insights presents a non-trivial challenge, the implications of which will be examined in subsequent chapters of this thesis.

Chapter 4

A Risk-Based Framework for the Empirical Evaluation of Privacy Metrics

In the previous chapter, we explored methodologies for quantifying privacy leakage risks specifically within the context of synthetic tabular data. The approaches proposed in [49, 99] illustrate distinct computational paradigms. On the one hand, attack-based metrics quantify specific adversarial risks arising from access to synthetic data, such as linking individuals, singling them out, or inferring sensitive attribute values. On the other hand, similarity-based metrics evaluate similarity between datasets, predicated on the principle that synthetic data must not exhibit excessive similarity to the original records according to a predefined distance metric.
However, these specific methods represent only a fraction of the broader landscape. The field currently lacks a universal standard for quantifying privacy risks, leading to fragmented and inconsistent approaches in practice. This fragmentation is worsened by privacy's inherently interdisciplinary nature, which spans complex legal, technical, and ethical dimensions. Consequently, practitioners often face significant challenges in selecting and justifying the appropriate privacy metrics for a given use case, particularly given the overwhelming variety of available tools and the lack of clear guidance. Furthermore, the practical difficulties of aligning technical metrics with regulatory compliance requirements further complicate this task, creating a bottleneck that can limit innovation in data-driven fields [43].
These challenges highlight an urgent need for a comprehensive analysis and a standardized framework to organize and contextualize privacy assessments—a need that has been widely identified in prior literature [8, 9, 13, 109, 111]. To address this, this chapter explores the state of the art in privacy metrics, proposing a comprehensive taxonomy that categorizes these metrics and establishes critical links to the relevant legal background. The findings and classification structure presented in this chapter are derived from the work originally published in [98]. The main contributions are:

- **A Unified Taxonomy**: We propose a comprehensive taxonomy that categorizes privacy quantification methods into Privacy Properties, Statistical Indicators, and Attack Simulations, explicitly mapping them to legal definitions (GDPR/WP29) to bridge the gap between technical metrics and compliance.

- **Risk Model Framework**: We introduce a novel experimental framework based on "Risk Models" (Leaky, Overfitting, and Differential Privacy) that allows for the systematic, empirical verification of a metric's sensitivity to actual data leakage.
- **Differentiation of General vs. Specific Inference**: We operationalize the legal distinction between data that "relates to" an individual via content versus purpose by implementing robust baselines (Control Set and Canary Records), ensuring that metrics measure specific memorization rather than general statistical learning.
- **Empirical Convergence**: Through extensive testing, we demonstrate a strong correlation between complex uniqueness-based attacks and simpler distance-based statistical indicators (like DCR) under no-box conditions, suggesting that computationally efficient indicators are often sufficient proxies for complex adversarial simulations.

4.1 The concept of privacy

The concept of privacy has already been briefly discussed in Section 2.2.1. The GDPR [38] introduces the concept of anonymization and identification of an individual. In particular, any individual in an anonymized dataset should no longer be identifiable. The document also states the principle of reasonableness and proportionality in Recital 26 [39], for which the identification process is defined. Beyond that, there are no specifications for how the data must be anonymized or what the anonymization process should look like. However, the Article 29 Data Protection Working Party (WP29) [5] selected the three attacks for which data should not be susceptible: Singling Out, Linkability, and Attribute Inference. Regarding the last attack, many considerations can be made. Attribute Inference Attacks (AIAs) are based on the prediction of sensitive attributes using information from other generated attributes. Knowing that there is an intrinsic correlation between attributes, it is difficult to state whether we have an actual privacy risk or if it is just a consequence of the natural correlation between attributes [62, 84].

Alternative formulations of privacy were made because the WP29 definition does not align with the technical classification of privacy or re-identification risk. In [66, 109] the risk is classified in two categories:

- *Identity disclosure*: given a set of known characteristics, it is the ability of identify an individual.
- *Attribute disclosure*: given a set of known characteristics, it is the ability to determine the unknown attributes of a record in the original dataset.

This classification aligns with the concept of Singling Out and Inference from WP29, not mentioning Linkability. In Raab et al. [109], the definition focus on what can be inferred from an information release, which can be thought of as either the individual identity or one value of its attributes. Moreover, the concept of *auxiliary information* is provided, highlighting its importance as an adversary can exploit it to create attacks. These two remarks clarify that the Linkability attack, technically, can be thought as an attack that exploits auxiliary information.

Besides these definitions, some known attacks on machine learning models are also used in the field of synthetic data generation; for example, adversarial attacks [74]. Following this intuition, the main risks for synthetic data generation are:

- Membership Inference Attacks (MIAs): the attacker (or adversary) tries to deduce whether a record was in the training set of the given machine learning model.
- Shadow Modeling: the adversary can reproduce the input-output pair of a machine learning model (in this case, the synthetic data generator).

These considerations are formalized in [123], where the authors formalize the WP29 attacks through MIAs. This last approach needs to define also the way an attacker can access to both the auxiliary information and the machine learning model itself, again, the generator. In the following section we identify different ways to access them.

4.2 Formalization of the Threat Models

A *Threat Model* identifies all the resources that an attacker can exploit to execute an attack. In general, those resources comprise the synthetic dataset, generator access, and auxiliary information. Regarding generator access, we have three different scenarios: *No-box generator access*, *Black-box generator access*, and *White-box generator access*.

No-box generator access

In this setting, the adversary has no access to the generative model in any way, apart from having a synthetic dataset generated from it. No other information, such as the architecture, is employed. The attacker can use auxiliary data from other sources to create the attacks. The auxiliary information is formalized in Definition 4.2.1.

Definition 4.2.1. *(Auxiliary information).* Let $\mathcal{D}$ be a dataset with attributes $\mathcal{A}(\mathcal{D})$. An adversary has **auxiliary information** if there exists a subset $A \subseteq \mathcal{A}(\mathcal{D})$ of attributes and a subset $\mathcal{D}' \subseteq \mathcal{D}$, such that the adversary knows the values $v(d, A)$ for all $d \in \mathcal{D}'$. We refer to the attributes in set A as **quasi-identifiers**. We denote auxiliary information consisting of quasi-identifier set A and records $\mathcal{D}' \subseteq \mathcal{D}$ by $\textsc{Aux}(A, \mathcal{D}')$.

Black-box generator access

In black-box generator access, the attacker has access to the trained synthetic data generator or the generator and its training procedure. In this way the attacker can use the trained model to generate how many synthetic record it needs, or it can train the generator on any dataset matching the algorithm capabilities.

White-box generator access

The attacker has access to the trained model, its architecture, the training algorithm, the hyperparameters, and the model's internal routines. The conditions to carry out white-box or even black-box attacks are typically not met in practical scenarios [104].

In the following work we will focus on the No-box generator access, that is, the attacker will only have access to the synthetic dataset and some auxiliary information.

4.3 Taxonomy for Privacy Quantification

We focus our taxonomy for privacy quantification around three approaches:

1. Privacy properties.
2. Statistical privacy indicators.
3. Attack simulation.

Figure 4.1 provides a schematic overview of the approaches for privacy risk quantification according the first categorization we made.

4.3.1 Privacy Properties

Privacy properties constitute the first major approach in our taxonomy, focusing on establishing formal, quantifiable guarantees on the data or the release mechanism itself. These methods define the acceptable limits of privacy loss a priori, before data is released or analyzed, differentiating them from post-release statistical or adversarial measurements. This category encompasses definitions rooted in structural obfuscation, exemplified by k-anonymity, and those based on algorithmic stability, most notably Differential Privacy.

Differential Privacy

Differential Privacy [32], as already seen in Section 2.4, is a property of an algorithm that limits the importance (or influence) of any point in the training dataset through the privacy budget ε. Differential privacy introduces noise during the learning phase, making the presence of a specific individual less important. This method makes more difficult for an attacker to target a specific record. the privacy budget is specified by the user, so that it can control the level of privacy protection. A low value of ε (e.g., a value that ranges from 1.0 and 5.0) indicates a high level of protection. Conversely, a high value of ε indicates a low level of protection. Moreover, as already stated in Section 2.4.1, this framework provides many mathematical properties that make Differential Privacy one of the most used frameworks in practice [44, 46, 69, 89, 114, 127].
In contrast to the metrics we have seen previously (e.g., Anonymeter attacks or DCR), which quantify privacy a posteriori, when the generator is trained and used to generate a synthetic dataset, DP is a property of the generator defined a priori with its parameters. Therefore, DP is included as a privacy quantification method because of its strong connection to the amount of information leaks that is controlled by the privacy budget. Despite this, DP can also be considered a risk model, such as the noisy synthesizer or the overfitting experiment, because, through the selection of ε, we control the amount of information leakage in our synthetic dataset.

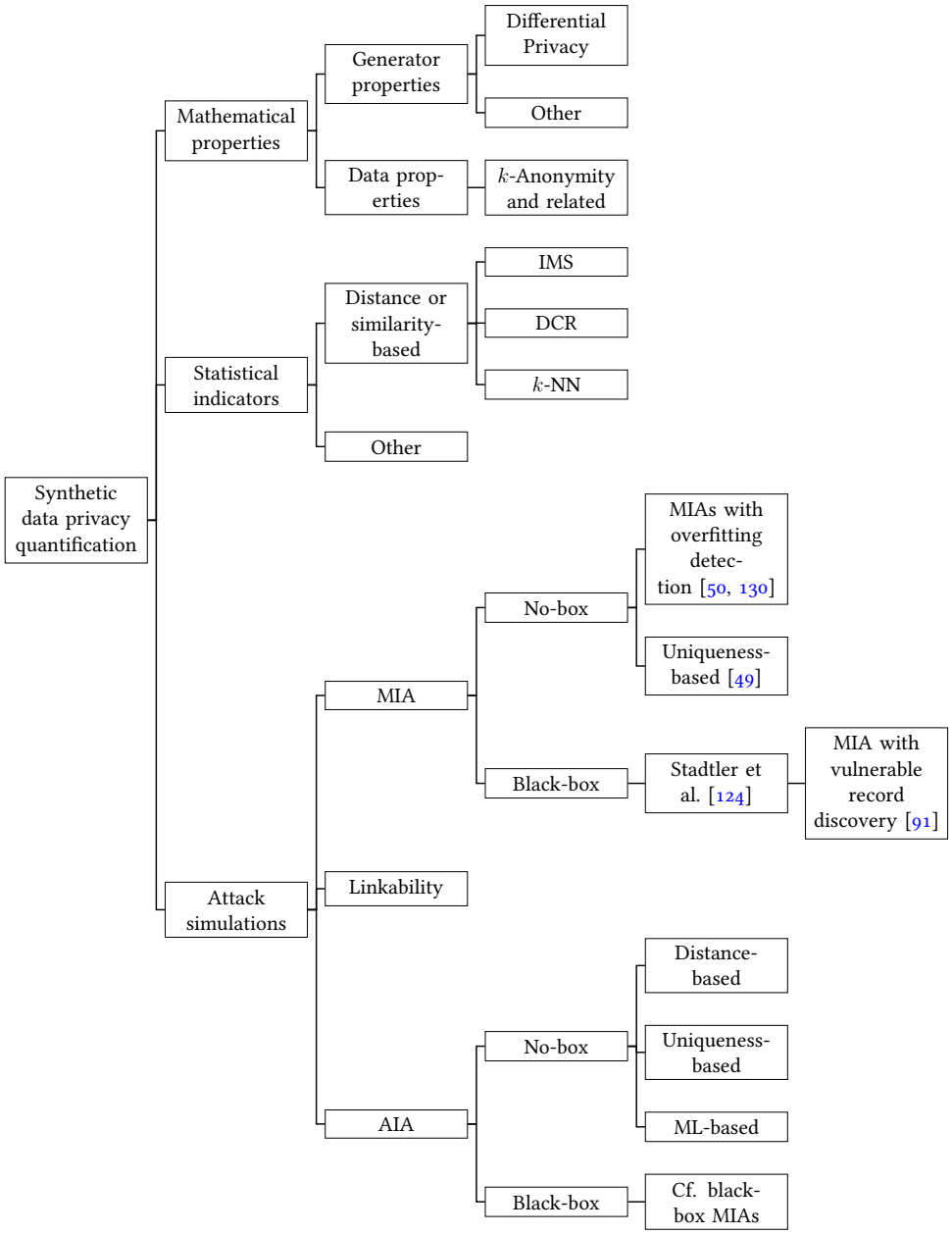


Figure 4.1: Overview of synthetic data privacy quantification methods

k-Anonymity

Another important property is the k-Anonymity, which has already been reviewed in Section 2.2.1. In this case, k-Anonymity is a property of the dataset and not of the generator as DP. The main concept for this property is that the uniqueness of an individual is the principal risk to be considered when releasing a dataset. Generalization and other techniques are used to hide highly specific values that would make identification easy for an attacker. For example, if in a census dataset, there is only one individual who lives on a specific street, then knowledge of the attribute "residence" is sufficient to unambiguously identify that individual. Generalizing that attribute to "City" or even "Region" provides good privacy protection from the risk of identification. The concept of k-Anonymity formalizes this form of protection by grouping individuals into groups, or equivalence classes, of size k, where every individual is indistinguishable from the others within it. The complete formalization of the approach is described in Section 2.2.1, where the concept of quasi-identifiers and other types of techniques used to achieve k-anonymity are also presented. But this type of procedure, intended as a privacy protection technique, is not able to protect against AIAs [5]. In fact, if an attacker has access to common knowledge, e.g., a particularly high income for a large group in the population, then even if we apply many techniques to achieve k-anonymity, the attacker can infer the income within a low margin of error. To address this problem, other techniques were proposed, such as l-diversity or t-closeness, both of which have already been explained in the previous chapters. The downside of using these approaches, despite their increasing privacy protection, depends on the strict conditions they require, which make them less appealing than k-anonymity. Leaving aside privacy protection, these 3 methods can be used as a privacy quantification technique. In particular:

- For k-anonymity, the parameter k serves as a direct probability cap. If a record is k-anonymous, the probability that an attacker can successfully link a specific record to a specific person is at most $1/k$

- For l-diversity, the parameter l quantifies the adversary's confidence. If a group is l-diverse, an attacker needs to eliminate at least $l-1$ other possibilities to be certain of a sensitive value. Then, If there are l distinct values with equal frequency, the probability of an attribute inference attack succeeds with probability at most $1/l$.

- For t-closeness, the risk is quantified as information gain. The variable t limits how much new information an attacker learns about a specific group compared to what they already knew about the general population. A lower t means the group looks more like the "average" population, revealing less specific information.

In essence, these frameworks quantify privacy risk by establishing statistical thresholds that strictly cap the probability of successful re-identification or sensitive attribute inference. They transform vague privacy concerns into measurable mathematical problems, limiting the information gain an adversary can extract by analyzing the probability distributions of the data.

4.3.2 Statistical Privacy Indicators

Statistical privacy indicators represent the second class of metrics in our taxonomy, bridging the gap between theoretical properties and adversarial attack simulations. These indicators quantify privacy protection by measuring the statistical proximity between the synthetic dataset ($\hat{\mathcal{D}}$) and the real training data ($\mathcal{D}$).
The core mechanism involves establishing an empirical baseline by analyzing the natural distribution of distances within the original data, often by comparing samples from the training set against a holdout set. This baseline defines the "statistically reasonable" level of proximity expected in the true data distribution. Any synthetic record found to fall outside this expected distance distribution, typically identified through nearest-neighbor analysis or quantile comparison, is flagged as a potentially hazardous outlier. The primary risk quantified by these methods is record memorization (or near-copy disclosure), where the generative mechanism replicates specific input instances. These indicators generally utilize distance-based statistics to assess a model's fidelity at the individual record level. Some examples and their methodological details will be examined in the following paragraphs.

Identical Match Share

Identical Match Share (IMS), also known as Replicated Uniques [108], repU [109], or Unique Exact Match [127], computes the portion of synthetic records that are exact copies of records of the training set [63, 100, 127]. Given the training dataset $\mathcal{D}$ and the synthetic data $\hat{\mathcal{D}}$, IMS is computed as:

$$\text{IMS}(\mathcal{D}, \hat{\mathcal{D}}) = \frac{|\mathcal{D} \cap \hat{\mathcal{D}}|}{|\hat{\mathcal{D}}|} \tag{4.1}$$

The choice of the synthetic dataset's length ($|\hat{\mathcal{D}}|$) in the denominator for the Identical Match Share is crucial because the metric is designed to quantify the release risk, that is, the danger inherent in the specific data being made public. The resulting IMS value represents the proportion of the generated output that is compromised by being a copy of a training record. This correctly normalizes the risk with respect to the size of the released data, and this approach is particularly useful when the synthetic data is oversampled ($|\hat{\mathcal{D}}| > |\mathcal{D}|$), as the metric remains a direct measure of the compromise ratio of the released data.

Distance to Closest Record

Distance to Closest Record (DCR) has been already defined in 3.3.4. This metric extends the IMS by considering also quasi-identical records as possible responsible of privacy leaks. In this case, Synthetic-to-Real and Real-To-Real distances were introduces to compute the similarity between the synthetic records and the training ones and a threshold that identifies how much a synthetic record should be similar a real one to have a privacy leak. By extending IMS in this way, synthetic records that differ by one entry from a training record can put at risk individuals private information. Conceptually, DCR serves as a non-adversarial proxy for a re-identification threat model. The inequality

established by the SRD and RRD comparison provides empirical evidence that the synthetic record in question is suspiciously close to a training record, indicating a failure to generalize. This proximity is the fundamental signal of potential record memorization by the generator. To conclude, the efficacy of DCR hinges entirely on the choice of the similarity/distance metric used. Since tabular data often contains mixed categorical and numerical attributes, the metric must accurately measure similarity across these disparate types (e.g., utilizing a weighted Euclidean distance or a composite metric like Gower distance). A poorly chosen distance function will invalidate the RRD baseline and render the DCR score meaningless.

k-Nearest Neighbor indicators

DCR can be generalized by considering more than the first neighbor. k-Nearest Neighbor (k-NN) indicators compare the neighborhood of synthetic and training records. We define the synthetic to real k-neighborhood of a synthetic record $\hat{d} \in \hat{\mathcal{D}}$, denoted by $N^k_{\text{SRD}}(\hat{d})$ as the set of k nearest records to $\hat{d}$ in $\hat{\mathcal{D}}$ with respect to the SRD. We define the real-to-real k-neighborhood $N^k_{\text{RRD}}(d)$ analogously using the RRD. We then apply equation (3.11) with the means of the neighborhoods $N^k_{\text{SRD}}(\hat{d})$ and $N^k_{\text{RRD}}(d)$ in place of the SRD and RRD.

4.3.3 Attack Simulation

We include attack simulations described in the WP29. All three of these risks can be implemented with different attack mechanisms. Below, we provide an overview of possible risk mechanisms and our implementations.

Membership Inference Attacks

In Membership Inference Attack (MIA), the attacker tries to infer whether a record was used in the training set for the synthetic data generator or not [74]. MIAs are also known for their ability to model Identity Disclosure [49, 91, 130]. In this context, we can identify two different ways of using MIAs depending on the threat model. On one hand, we have a **No-box** generator approach in which we have only access to one generated dataset. On the other hand, we have a **Black-box** approach that require the training algorithm of the generator to create different generator models that are trained on different instances of the training dataset.

In the **No-box** approach we can distinguish two different methodologies. The former is the **Uniquenes-based MIA**, in which the attacker detects vulnerable records in the synthetic data and identify them as potential original records. There are many ways of identifying vulnerable records that are mainly based in finding the outliers of a dataset. Knowing that synthetic datasets are usually a mixed-datatypes objects, the outlier search is done after a preprocessing step where the dataset's attributes are entirely transformed into numeric attributes. In [99] (our implementation of Singling Out with contrastive learning), the outliers are searched after this step using Local Outlier Factor [15]. In [91], vulnerable records are found by looking for the records with the highest distance to their k-nearest neighbors. The choice of the distance metric makes the preprocessing step

either mandatory or not. In [49] (Anonymeter's Singling Out), the vulnerable records are identified by looking for uniqueness in categorical attributes and extreme values for numeric ones.
The second approach involves **Overfitting Detection**. Prior methods to DOMIAS [24, 56, 58], rely on only synthetic data and the estimate of its density on the set of test points to infer membership. This methodology has been proved to be inadequate because it does not distinguish between overfitting in the generator or a genuine density peak in the dataset. That's why DOMIAS [130] extend this approach by considering also the real distribution of the data. In particular, the original dataset $\mathcal{D}$ is partitioned into a training set $\mathcal{D}_{\texttt{train}}$, a control set $\mathcal{D}_{\texttt{control}}$, and a reference dataset $\mathcal{D}_{\texttt{reference}}$. A generative model $G(\cdot)$ is trained on $\mathcal{D}_{\texttt{train}}$ to produce a synthetic dataset $\hat{\mathcal{D}} \sim G(\mathcal{D}_{\texttt{train}})$. The idea of this attack is that the generator overfits all the points in the training set, especially the outliers, consequently, those points show a higher density in the synthetic data distribution than in the true underlying distribution. The test set is composed by points from both the training and the control set. It is important to notice that we will use just a subset of those two sets, in particular, we will target the outliers of the training set, ignoring the points that lie on high density region, while we will sample at random the points from the control set in order to have $|\mathcal{D}'_{\texttt{train}}| = |\mathcal{D}'_{\texttt{control}}|$, where we indicate with the apostrophe that we are taking a subset. Once we have defined the data we will use within this attack, we must estimate the densities of both the synthetic and the release data. Two methods where tested, namely: Kernel Density Estimator (KDE), and Block Neural Autoregressive Flow (BNAF) [29]. DOMIAS is computed as it follows:

$$A(x^*) = f\left(\frac{p_{\texttt{synth}}(x^*)}{p_{\texttt{reference}}(x^*)}\right) \tag{4.2}$$

where $p_{\texttt{synth}}$ and $p_{\texttt{reference}}$ represents the learned distribution of the synthetic and reference data respectively, x^* is the target point, and $f : \mathbb{R} \to [0, 1]$ is a monotonically increasing function. If x^* is an outlier from the training set, the numerator dominates the denominator because the point will be overfitted by the generator. Conversely, if x^* belongs to $\mathcal{D}_{\texttt{control}}$, the numerator and denominator yield approximately equal values. Finally, to compute the score for this attack, the area under the ROC curve (AUC-ROC) is computed for evaluation.

In the **Black-box** approach, we can identify many different methodologies. In shadow modeling, the attacker access the training algorithm of the generative model and train $2k$ generator models in the following way:

1. The adversary takes k datasets $\mathcal{D}_1, ..., \mathcal{D}_k$ with $\mathcal{D}_i \subset \mathcal{D} \setminus \{d_t\}\,, \quad i = 1, ..., k$;
2. For each such dataset $\mathcal{D}_i$, the adversary trains a generator $G(\mathcal{D}_i)$ and produces an output dataset $\hat{\mathcal{D}}_i \sim G(\mathcal{D}i)$. The adversary stores datasets $\hat{\mathcal{D}}_i$ along with the label "no target";
3. Next, the adversary creates the k datasets $\mathcal{D}'_i := \mathcal{D}_i \cup d_t$ for $i = 1, ..., k$, obtained by adding the target record to the chosen datasets;
4. The adversary now trains generators $G(\mathcal{D}'_i)$, producing synthetic datasets $\hat{\mathcal{D}}'_i \sim G(\mathcal{D}'_i)$. The adversary stores the sets $\hat{\mathcal{D}}'_i$ along with the label "target";

5. The adversary now has a labeled collection of synthetic datasets. They leverage this collection to train a model $\mathcal{M}$ that predicts whether the target record d_t was in the dataset used to train a generator G, based on an output synthetic dataset of G.

Usually, in Step (5) are used machine learning models, such as XGBoost or Neural Networks, to classify whether that specific record was in the training set or not. To facilitate the learning phase, feature extraction algorithm can be used to extract specific information from each dataset in order to use lighter models for the classification. The described procedure is highly computationally demanding because it has to repeat this procedure for every record in the test set, which ends up in training $2kn_{\text{test}}$ times the synthetic data generator which can take weeks to complete. This is essentially why other methods are preferred for computing MIAs.

Linkability

To the best of our knowledge, Anonymeter's Linkability attack [49] is the only direct no-box implementation of linkability attacks. The procedure is already explained in Section 3.2. First, the original dataset is split in two parts vertically, so that the splits contain different attributes. Then, the attacker tries to correctly link the split records using the synthetic dataset as knowledge looking for the nearest neighbors between the split and the records in the synthetic data. If the intersection of the neighborhoods of the two splits is not the null set, then the attack is considered successful.

Attribute Inference Attacks

In Attribute Inference Attacks (AIAs), the attacker tries to correctly infer a sensitive attribute of a real individual using the available information on the other attributes. Like identity disclosure attacks, many mechanisms can be used to perform a correct inference. Given a set of known attributes A and the synthetic dataset $\hat{D}$, the attacker will use a model $\mathcal{M}$ to correctly infer the target attribute t as follows:

$$\hat{t} = \mathcal{M}(\hat{\mathcal{D}}, A) \tag{4.3}$$

Again, according to the different implementation of the model $\mathcal{M}$ we can identify different types of AIAs.

Uniqueness-based AIAs

In Uniqueness-based AIAs, the attacker tries to infer the target attribute t by isolating unique records in the synthetic dataset $\hat{\mathcal{D}}$. Usually, this is done by identifying equivalence classes, e.g., a set of records that share the same values for the quasi-identifiers. In this setting, the **Targeted Correct Attribution Probability** (TCAP) [128] provides the probability of having a correct inference by using the synthetic dataset and the set of known quasi-identifiers A. TCAP is computed as follows:

$$\text{TCAP}(\mathcal{D}, \hat{d}) := \frac{\sum_{d \in \mathcal{D}} \Big[v(d, A) = v(\hat{d}, A),\ v(d, t) = v(\hat{d}, t) \Big]}{\sum_{d \in \mathcal{D}} \Big[v(d, A) = v(\hat{d}, A) \Big]} \tag{4.4}$$

where d denotes a generic record of the train data, analogously $\hat{d}$ is a generic record of the synthetic dataset. With $v(d, A)$ we identify the specific values of the attributes in the set A of the d row, $v(d, t)$ denotes the value of the target attribute in the d record, and the square brackets are the Iverson brackets. To understand the way TCAP works, consider a dataset that has been anonymized using k-anonymity. If the identified equivalence classes have a low l-diversity, that is, a low number of distinct values for the target variable, then the attribute inference attack is more powerful. This vulnerability is measured by the TCAP in a probabilistic way. The formula 4.4 computed this probability by calculating the ratio of records between the training and synthetic datasets that share the same combination of values of quasi-identifiers and the target variable and records that only share the same values of quasi-identifiers. The formula is thought of for categorical variables only, because we cannot ask for equality when numeric variables, especially real values, are considered. The generalization of TCAP (GTCAP) has been introduced in [22]. For numeric attributes, the equality is replaced by set membership as follows:

$$v(d, a) \in \left[v(\hat{d}, a) - r, v(\hat{d}, a) + r\right] \tag{4.5}$$

where r indicates the radii of the interval centered in $v(\hat{d}, a)$. It is important to note that the value of r depends on the scale of the attribute we are considering. A common strategy to address this is to preprocess the numeric variables by rescaling them and use the same value for r.

Distance-based AIAs

In distance-based AIAs, the attacker exploits similarities between synthetic and real records from the auxiliary information to achieve a correct inference on the unknown attribute t. In the literature (see [28, 49, 51, 62]), this type of AIA takes advantage of the available information at first by computing the distance between the partially known real record and the rows in the synthetic dataset. Once these distances are found and sorted in ascending order, the attacker will estimate $\hat{t}$ as $v(\hat{d}, t)$, in the case where we are considering only the first neighbor, otherwise the estimate $\hat{t}$ will be computed as the mean/median of the target values of the first k neighbors if the target variable is numeric, otherwise the mode will be used. In practice [27, 51], the best attack performance is achieved when only the first neighbor is considered ($k = 1$).

ML-based AIAs

AIAs based on machine learning techniques use synthetic data to train a machine learning model. The attacker then, along with auxiliary information, predicts the unknown attribute using the learned model Equation 4.3. Depending on the type of the target variable, we may have a classifier for categorical variables or a regression model for numeric variables. The selection of the model is up to the user but usually XGBoost model [25] is preferred because of its speed and performance.

Black-box AIAs

In Black-box AIAs, the attack is conducted as a sequence of MIAs [124]. Suppose that we have access to some auxiliary information like in the previous scenarios and we have to

guess the value of the secret attribute t. We denote the set of all possible values for the target attribute t as T . Then, we can run $|T|$ times a MIA by replacing at each iteration the value of t. If the record d is in the training data according the MIA, then we can conclude that the value of t in that record is the value we were looking for.

4.3.4 Distinguishing Specific from General Inference

The "Relating to" Standard in GDPR

A central challenge in determining whether synthetic data is truly anonymized is understanding when data "relates to" an individual under the GDPR. Legal scholarship, such as that by Cesar [83], distinguishes between three ways data can relate to a person: *in content*, *in purpose*, or *in result*.

- **Relating to in content:** This is the most direct form, where data contains specific facts about an identifiable person (e.g., "Jan Jansen is poor"). This is unequivocally personal data.
- **Relating to in purpose:** This occurs when data is used to evaluate or influence specific individuals, even if the data itself is not directly identifying. For example, if a synthetic profile is used to target ads at "diabetic users in postcode 1234," and this targeting affects Jan Jansen, the data relates to him "in purpose" because it is being used to alter his environment or choices.
- **Relating to in result:** This refers to the downstream impact. If a model trained on real health data leads to insurance price hikes that affect Jan Jansen, the data relates to him "in result," even if his specific records were not in the training set.

These distinctions are critical because "purpose" and "result" often stem from general, population-level insights rather than specific data leaks. Treating general statistical accuracy as a privacy breach risks overextending the GDPR. A synthetic dataset should reflect population statistics; however, it becomes a privacy risk when it overfits—memorizing specific, unique details ("in content" data). To operationalize this, we adopt baselines from Giomi et al. [49] and Sundaram et al. [4] to separate general patterns from specific memorization.

Anonymeter control baseline

This formulation already has been explained in Section. 3.2, specifically in Equation 3.1. We employ the control set baseline to filter out risks attributable to general population patterns. By calibrating the attack efficacy against a hold-out set ($\mathcal{D}_{\text{control}}$), the resulting metric measures only the information specifically memorized from the training set.

Canary record baseline

Houssieau et al. [62] and Sundaram et al. [4] highlight the "**base-rate problem**" in Attribute Inference Attacks (AIAs). High attack success often stems from capturing general population correlations (e.g., the link between education and wealth) rather than from memorizing specific records. To decouple general patterns from specific information

leakage, we adopt the canary record baseline proposed by Sundaram et al. [4].

Let $\mathcal{D}$ be a dataset and t a target attribute. We define a canary record d' by randomizing the target attribute of an existing record d, as shown in Equation (4.6):

$$v(d', a) = \begin{cases} v(d, a), & \text{if } a \neq t \\ r, & \text{if } a = t \end{cases} \tag{4.6}$$

where r is drawn uniformly from the domain of t. Because r is random, it is uncorrelated with the remaining attributes in d'. Consequently, if an AIA successfully infers the value r from a generator trained on the modified dataset $\mathcal{D}'$ (where d is replaced by d'), the success cannot be attributed to general correlations. Instead, it indicates that the generator has memorized the specific content of record d'. In our evaluation, we average AIA success rates across 100 randomly selected canary records per dataset.

Summary of Theoretical Claims and Experimental Validation.

The taxonomy and threat models defined in Sections 4.1 to 4.3 establish a structured language for privacy quantification. Our theoretical development proposes that Risk Models (Leaky, Overfitting, and Differential Privacy) can serve as reliable, monotonic proxies for ground-truth vulnerability, allowing for the objective benchmarking of metric sensitivity in the absence of a real adversary. Furthermore, we hypothesize a convergence of efficacy among No-box metrics; specifically, that under strict black-box conditions, computationally expensive attack simulations (like those in the Taxonomy's third pillar) and simpler statistical indicators (like DCR from the second pillar) measure fundamentally similar phenomena of local memorization.
In the following experimental evaluation, we validate these hypotheses by subjecting a wide range of metrics from the taxonomy to the proposed Risk Models. We aim to empirically demonstrate which metrics provide the most reliable "signal-to-noise" ratio in detecting specific leaks and to determine if simple statistical indicators can serve as sufficient proxies for complex attacks in routine auditing scenarios.

4.4 Experimental Evaluation

To empirically evaluate the performance of privacy quantification methods, we introduce the concept of a risk model. We define a risk model as a mechanism for systematically injecting privacy vulnerabilities into synthetic datasets to measure a method's ability to detect them. In the following section, we present the risk models used in our experiments, detailing their rationale and specific parametrization.

4.4.1 Risk models

We evaluate the sensitivity of privacy metrics by employing three distinct risk models that introduce vulnerabilities in a controlled manner. These approaches range from direct data leakage and induced generator overfitting to the systematic relaxation of Differential Privacy budgets.

Leaky risk model

The leaky risk model was already presented in 3.2, where we refer to it as the Leaky Synthesizer. Briefly, we create 3 non-overlapping sets from the original dataset, namely: train, control, and release. We create the synthetic dataset by combining the records from both the train and the release datasets according to a leak fraction l_f that controls the portion of training records present in the synthetic dataset. Incremental risk insertion allows us to assess whether the metrics scale linearly with added risk.

Overfitting risk model

The Leaky model provides a controlled environment, but it does not provide a realistic example. Privacy risks emerge during the training of the synthetic data generation model; in particular, overfitting has been detected as the main cause of the risk [13, 19, 124]. Overfitting of the generator model may lead to training data memorization, so its output would not be stochastic as expected but would exhibit a "copy & paste" behavior. The overfitting risk model has already been introduced in Section 3.3.5, in particular, we train the synthetic data generator at different levels of overfit according to the overfit ratio defined in Equation 3.13. In this way, we are mimicking the behavior of the leaky risk model but using a real synthesizer.

Differential Privacy

The last approach focuses on Differential Privacy generators. As already presented in Section 2.4, the DP generators learn as a common SDG with the addition of some hyperparameters. In particular, we will try to control the private information injection in the synthetic dataset by generating many synthetic datasets at different levels of privacy budget ε. We will evaluate the metrics using $\varepsilon \in [1.0, 5.0, 10.0, 50.0, 100.0]$ so that we can mimic leaky and overfitting risk models by increasing the privacy budget in order to nearly disable the DP mechanism.

4.4.2 Other Experiments

To conclude our experimental evaluation, we propose other experiments to assess the robustness of the privacy quantification methods we will study.

Outliers Removal

Privacy quantification methods may be sensitive to specific data properties. In this experiment, we will focus on the presence of outliers [20, 64, 124] combining threat models with outlier removal to demonstrate this vulnerability. We will repeat the measurement of all the metrics after removing a certain percentage of outliers. In particular, we will use the method proposed in [99], so at first we will use the presented contrastive learning method to learn a numeric representation of our dataset, then we will use Local Outlier Factor to compute the outliers according the defined portion. In our experiments, we will remove the following portion of outliers: 1%, 2%, 5%, and 10%. We repeat the experiment using the overfitting risk model, in particular, we will take into account only the

overfit ratio of 1.0 and 1.6 To combine the impact of removing outliers with intentional overfitting.

Hyperparameter Sensitivity

We also investigated other parameters regarding some specific methods due to their susceptibility. In particular, we will focus on:

1. The number of neighbors k in the DCR
2. The radii of the GTCAP

Even if some previous work has already made this study (see [28, 51] for the former experiment), we believe that repeating the experiment would confirm previous findings. With respect to the GTCAP radii, we try to increase the level of awareness on the importance of the hyperparameter selection and that it might be dependent from the context and, specifically, on the dataset.

Synthetic data size

To determine if the size of the dataset significantly influences our evaluation metrics, we will generate a synthetic dataset with systematically varied sizes (number of records). This controlled experiment is designed to investigate the metrics' dependence on the volume of synthetic data. By comparing the metric values calculated across these differently sized datasets, we can directly observe if the metrics are susceptible to cardinality changes, confirming whether dataset size is a non-negligible factor in their stability and reported performance.

4.4.3 Metrics

The metrics used in this study are presented in Table 4.1. We evaluate all metrics using the risk models detailed in Section 4.4.1. For the leaky risk model, we vary the leak fraction l_f from 0 to 1 in increments of 0.2. Similarly, for the overfitting model, we vary the overfit ratio f_o from 1 (no overfitting) to 2 (where $L = 2L^*$) in increments of 0.2. Finally, to assess DP risk, we utilize OpenDP Smartnoise AIM [89] and PATEGAN [69] to train ε-DP synthetic data generators using privacy budgets $\varepsilon \in \{1.0, 5.0, 10.0, 50.0, 100.0\}$.

Statistical Indicators

The IMS implementation is straightforward. The 2nd percentile is used for both DCR and k-NN. We calculate Euclidean distances within an embedded space derived using the methods described in [100]. To calibrate the metric, we normalize the DCR relative to its theoretical best and worst cases as described in Section 3.3.4. Consequently, the score scales from 0 (indicating no information leakage or perceived risk) to 1 (indicating that all synthetic records pose a risk to the original data). To estimate the variance of the methods, we apply bootstrapping with $n = 1000$

Table 4.1: Synthetic data degree of privacy quantification methods used in this study. Risk: risk measured or controlled; Aux: auxiliary information; MIA: membership inference attack; AIA: attribute inference attack; disc.: disclosure; SO: Singling Out; Link: Linkability; ML: machine learning; IoS: inference-on-synthetic; LN: local neighborhood. Exactly the attribute disclosure attacks require access to auxiliary information. Nb: examples are for reference only: they may implement same attack methods and mechanisms in different manner than our implementations.

Method	Type	Risk	Mechanism	Aux.	Example(s)	Our implementation
Differential privacy	Generator property	Propensity	NA	No	[89, 114]	OpenDP AIM
Differential privacy	Generator property	Propensity	NA	No	[69]	Synthcity PATEGAN [107]
IMS	Statistical indicator	Propensity	NA	No	[63, 100, 108, 127]	Standard
DCR	Statistical indicator	Propensity	NA	No	[63, 100]	Percentiles-based
k-NN	Statistical indicator	Propensity	NA	No	[63, 100]	Percentiles-based
Outlier-based MIA	Attack	SO	uniqueness	No	[49] (SO), [91]	Anonymeter [49] SO
Outlier-based MIA (DOMIAS)	Attack	SO	uniqueness	Yes	[130]	ROC AUC classifier
Linkability attack	Attack	Link	distance	Yes	[49] (Link)	Anonymeter [49] Link
Classifier inference	Attack	AIA	ML	Yes	[62] (IoS)	XGBoost, accuracy
Regression inference	Attack	AIA	ML	Yes		XGBoost, RMSE
Distance-based AIA	Attack	AIA	distance	Yes	[49] (AIA), [62] (LN), [28, 51]	Anonymeter [49] AIA
GTCAP	Attack	AIA	uniqueness	Yes	[22, 27, 128]	See [22]

MIAs

For the uniqueness-based MIA, we employ the Singling Out attack from Anonymeter [49]. We conduct experiments targeting outliers in single attributes, as well as experiments varying the number of attributes between 3 and 12. Each run consists of a maximum 2000 attacks;* we report the highest observed risk R. For MIAs with overfitting detection, we adopt the approach introduced by van Breugel et al. [130]. The area under the ROC curve (AUC-ROC) is computed to evaluate the attack because of its nature as a binary classifier. The results are then normalized to restrict the measure in the interval $[0, 1]$ for a better interpretation.

AIAs

In our experiments, numerical data is normalized using a MinMaxScaler, ensuring values fall within the (0,1) interval; consequently, we set the GTCAP radius to 0.1. We assume a worst-case scenario where the adversary knows all attributes except the target. To assess distance-based AIAs, we utilize the inference attack from Anonymeter [49] using the authors' original parameters. For ML-based AIAs, we employ the default unregularized implementation of XGBoost from *Scikit-Learn*. We apply classification (evaluated via accuracy) for the Adult dataset, and regression (evaluated via RMSE) for the Census and Texas datasets. To align the regression metrics with classification accuracy, we apply a heuristic normalization to the RMSE. Specifically, we calculate the ratio of the RMSE to the data's total range and subtract this value from 1. This transformation results in a score where 1 represents a perfect prediction and values below 1 indicate increasing error (with values < 0 denoting performance worse than the data's inherent variability). While this allows us to plot accuracy and regression performance on a similar scale, we caution that this normalized metric is a heuristic and not strictly equivalent to accuracy.

$$\text{NRMSE}(y, \hat{y}) = 1 - \frac{\text{RMSE}(y, \hat{y})}{\text{range}(y)} \tag{4.7}$$

Regarding Linkability, we employ the default implementation provided by Anonymeter [49].

Comparative Analysis of Privacy Auditing Tools

To better contextualize the proposed framework within the current state of the art, we provide a comparative overview of the primary privacy auditing tools evaluated in this thesis. Table 4.2 summarizes the key characteristics of these frameworks, highlighting their specific threat models, strengths, weaknesses, and relative computational burden.

Utility

To ensure a comprehensive evaluation, we assess utility using Machine Learning Efficacy (MLE), a metric based on the indistinguishability of the data. Inspired by the discriminator concept in Generative Adversarial Networks [52], this approach measures how easily a binary classifier can differentiate between real and synthetic samples. We construct

*The actual number of attacks depends on the count of predicates that successfully single out unique synthetic records. Consequently, this number may be lower than 2000 if fewer vulnerable records are detected.

Table 4.2: Comparative Summary of Privacy Auditing Frameworks

Tool	Type	Threat Model	Strengths	Weaknesses	Burden
IMS	Similarity	Identity (Exact)	• Fast & simple. • Easy to interpret.	• Only exact copies. • Misses near-matches.	**V. Low**
DCR	Similarity	Identity (Prox.)	• Standard baseline. • Captures approx.	• Metric sensitive. • Curse of dim.	**Low**
Anonymeter	Attack	Singl. Out, Link, Inf.	• Legal/GDPR map. • Conf. intervals.	• Slow (Brute-force). • Rigid metrics.	**Med/High**
DOMIAS	Attack (MIA)	Member. Inf.	• Theory grounded. • Overfitting focus.	• Density est. issues. • High dependence on outliers.	**High**
GTCAP	Attack (Attr)	Attr. Inf.	• Theory grounded.	• Slow (Brute-force).	**Med/High**
ML Inf.	Attack (Attr)	Attr. Inf.	• Utility proxy. • Predictive power.	• Indirect measure. • Model dependent.	**Med**

balanced datasets containing equal numbers of real and synthetic records and train an XGBoost classifier [25] to distinguish between them. The MLE is defined as the resulting test accuracy: a score near 0.5 indicates high utility (the synthetic data is effectively indistinguishable from the real distribution), whereas a score approaching 1.0 implies low utility (the classifier easily identifies synthetic records).

4.4.4 Synthetic Data Generation Models and Datasets

To conclude the experimental settings, we illustrate the datasets and the models used. First, the datasets used are the same ad the one described in Section 3.3.5. For what concern the models, we use 2 synthetic data generators that employ differential privacy in the training phase, namely: AIM [89] and PATEGAN [69]; and 2 non-DP synthesizer, that is: REalTabFormet [122] and Synthpop [95]. All the models have been discussed in Chapter 2, specifically in Sections 2.2.3 and 2.4.2 apart from Synthpop.

Briefly, Synthpop uses a method called Sequential Regression Modelling (SRM) [110, 131] that works as it follows:

1. The ordered sequence of the attributes is found
2. Starting from the first attribute a_1 in the sequence, we random sample a value from the original column
3. To obtain the value for a_2 we sample from the conditional distribution $a_2|a_1$. This process is repeated for all the other attributes by correcting on which attributes we are conditioning our sample. For example, at step i, we will sample from the distribution $a_i|a_{i-1} \wedge a_{i-2} \wedge \ldots$.

One of the key strengths of Synthpop is its flexibility: you are not locked into a single algorithm. Through the method parameter in the syn() function, you can specify exactly which model to use for each variable, whether that is the default CART, parametric methods like linear or logistic regression.

4.5 Results

In this section, we present a summary of the experimental results, focusing on the comparison between no inserted risk and maximal inserted risk (Table 4.3) as well as the computation times for each metric (Table 4.4). Because the privacy quantification methods measure different quantities, comparisons should focus on the *response to inserted risk* rather than the exact normalized values. For the complete experimental evaluation, including full result tables and correlation matrices for all risk models, please refer to Appendix A.1.
The computation times demonstrate a clear trade-off between speed and complexity, with simple distance metrics (IMS, DCR) being consistently fast and stable across scenarios. In contrast, complex adversarial simulations like GTCAP and distance-based inference attacks are resource-intensive, with costs scaling significantly based on dataset size and dimensionality (Census).

Table 4.3: Results of the leaky and overfitting risk models. RTF: RealTabFormer; O: outlier; D: distance; ML: machine learning. By "no risk", we indicate that no risk was deliberately added, i.e., $f_l = 0$ for the leaky risk model; $f_o = 1$ for the overfitting risk model; for the DP risk model, we equate "no risk" to a privacy budget of $\varepsilon = 0$. Simliarly, "max risk" refers to $f_l = 1$; $f_o = 2$; and $\varepsilon = 100$. We use the asterisk (*) to denote the maximum risk, which exceeds any risk achieved with the previous values of f_l, f_o, or ε.

	Leaky			**Overfit**						**DP**					
				RTF			**Synthpop**			**PATEGAN**			**AIM**		
Method	**Adult**	**Texas**	**Census**	**Adult**	**Texas**	**Census**	**Adult**	**Texas**	**Census**	**Adult**	**Texas**	**Census**	**Adult**	**Texas**	**Census**
IMS (no risk)	0.0	0.0	0.0	0.0	0.0	0.0010	0.0	0.0	0.0037	0.0	0.0	0.0	0.0	0.0	0.0
IMS (max risk)	1.0	1.0	1.0	0.8684	0.0162	0.9428	0.4377	0.0	0.1519	0.0	0.0	0.0	0.0	0.0	0.0
IMS stdev (max risk)	0.0045	0.0038	0.0034	0.0034	0.0008	0.0017	0.039	0.0	0.0015	0.0	0.0	0.0	0.0	0.0	0.0
DCR (no risk)	0.0011	0.0082	-0.0013	0.0008	-0.0009	0.0046	0.0002	-0.0203	0.0064	-0.0204	-0.0204	-0.0204	-0.0057	-0.0204	-0.0186
DCR (max risk)	1.0	1.0	1.0	0.5325	0.4783	0.5700	0.2933	-0.0204	0.1115	-0.0204	-0.0204	-0.0204	0.0019	-0.0201	-0.0046
DCR stdev (max risk)	0.0001	0.0	0.0	0.0011	0.0010	0.0010	0.0032	0.0	0.0005	0.0	0.0	0.0	0.0008	0.0001	0.0005
MIA, O (no risk)	0.0060	0.0400	0.0101	0.0278	0.0310	0.0270	0.0148	0.0153	0.0285	0.0136	0.0213	0.0010	0.0146	0.0072	0.0137
MIA, O (max risk)	0.9990	0.9990	0.9989	0.7620	0.6744	0.7895	0.4836	0.0257	0.1025	0.0101	0.0239	0.0010	0.0627	0.1279	0.0318
MIA, O stdev (max risk)	0.0010	0.0010	0.0011	0.0204	0.0216	0.0298	0.0272	0.0126	0.0296	0.0173	0.0138	0.0024	0.0316	0.0210	0.0282
DOMIAS, O (no risk)	0.0040	-0.0066	0.0052	0.0966	0.0966	0.0120	0.0162	-0.0088	0.0074	-0.0014	-0.0016	-0.0054	0.0008	-0.0018	0.0
DOMIAS, O (max risk)	0.5368	0.0312	0.3152	0.3542	0.1444	0.2202	0.2154*	-0.0016	0.0136	0.0024*	-0.0012*	0.0034*	0.0076*	-0.0018*	0.0
Link. (no risk)	0.0015	0.0116	0.0030	0.0004	0.0111	0.0035	0.0015	0.0015	0.0065	0.0	0.0025	0.0005	0.0015	0.0010	0.0035
Link. (max risk)	0.6433	0.9934	0.6336	0.2854	0.3820	0.2874	0.1589	0.0065	0.0521	0.0005	0.0045	0.0010	0.0015	0.0015	0.0020
Link. stdev (max risk)	0.0211	0.0035	0.0213	0.0199	0.0218	0.0203	0.0161	0.0049	0.0103	0.0017	0.0059	0.0024	0.0025	0.035	0.0046
AIA, ML (no risk)	0.1570	0.1904	0.0052	0.0077	0.3370	0.0060	0.0887	0.6611	0.0	0.0	0.0	0.0023	0.0	0.6370	0.0
AIA, ML (max risk)	0.4499	0.9749	0.1535	0.3107	0.7193	0.1733	0.2175	0.4377	0.0337	0.0	0.0	0.0	0.0488	0.5702	0.0
AIA, D (no risk)	0.0835	0.1312	0.2153	0.0665	0.1262	0.2013	0.1086	0.0887	0.2159	0.0176	0.0177	0.0229	0.0731	0.0988	0.1720
AIA, D (max risk)	0.9922	0.9920	0.9579	0.5945	0.5557	0.6393	0.3621	0.1161	0.2857	0.0241	0.0229	0.0244	0.0923	0.1112	0.1968
AIA, D stdev (max risk)	0.0041	0.0080	0.0390	0.0551	0.0599	0.1221	0.0819	0.0887	0.1629	0.0585	0.0963	0.0908	0.1348	0.0466	0.1647
GTCAP (no risk)	0.0019	0.0005	0.0015	0.0089	0.0	0.0012	0.0817	0.0	0.0424	0.0	0.0	0.0	0.0933	0.0	0.0001
GTCAP (max risk)	0.9665	1.0	0.9897	0.5094	0.0156	0.8114	0.4513	0.0	0.1516	0.0	0.0	0.0	0.1303	0.0	0.0017

Table 4.4: Mean of computation times of the various measurements in seconds for the RealTabFormer and AIM models

Method	Leaky			Overfit			DP		
	Adult	Texas	Census	Adult	Texas	Census	Adult	Texas	Census
IMS	22.50	66.30	43.41	22.50	66.30	43.41	22.50	66.30	43.41
DCR	6.91	134.72	83.35	6.91	134.72	83.35	6.91	134.72	83.35
k-NN	6.91	134.72	83.35	6.91	134.72	83.35	6.91	134.72	83.35
DOMIAS (O)	2.11	46.45	68.45	2.21	45.71	65.45	2.90	47.86	71.45
MIA (D)	6.91	134.72	83.35	6.91	134.72	83.35	6.91	134.72	83.35
Link.	18.06	103.01	99.26	12.44	30.02	63.50	9.72	28.74	63.53
AIA (ML)	0.35	0.58	0.45	0.36	0.54	0.43	0.036	0.55	0.46
AIA (D)	163.23	1380.70	1603.49	74.98	438.33	1117.48	71.99	445.90	1278.39
GTCAP	417.05	2093.73	5580.93	420.77	2115.10	5488.23	435.16	2238.01	5789.22

Leaky risk model

In this setup, we will look at only the Adult plots for the metric we are considering because the results are analogous to the other datasets.

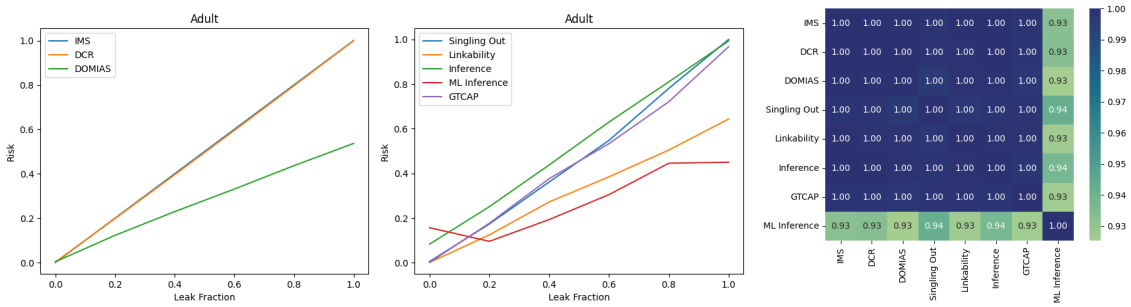


Figure 4.2: Metric comparison results on Adult dataset in the leaky risk model and correlation between the metrics

Under the leaky risk model, most privacy metrics show a highly desirable linear increase proportional to the leak fraction, typically reaching a score of 1.0. This indicates a strong and proportional response to the amount of direct data leakage, effectively showing that nearly all injected risk is successfully detected. As evidenced in Table 4.3, this general trend means that most metrics achieve or nearly achieve their maximum possible value when the leak fraction is at its maximum value. However, the Linkability attack, the ML-based Attribute Inference Attack, and DOMIAS are notable exceptions to this pattern. The Linkability attack does not consistently attain its maximum value because its efficacy relies heavily on the partitioning of attributes; if partial records contain many duplicates, linking remains difficult even with significant leaks. For the ML-based AIA, the behavior stems from the strong baseline performance of the underlying ML models on the control set, which inherently reduces the overall measured value of the metric. The performance of the DOMIAS attack is complex and determined by two technical factors: the density estimator's sophistication and the statistical separation between member and non-member data. A simple estimator may fail to detect the local density spikes caused

by model overfitting, leading to a low DOMIAS score due to poor separability. Conversely, if non-member points are highly similar to the training points, occupying the same high-density regions, the computed densities for both groups overlap significantly. This overlap, too, results in a low DOMIAS score, which is interpreted as the synthetic dataset offering strong privacy protection. Therefore, achieving a high DOMIAS score requires both a highly sensitive estimator and statistically discernible density differences between the member and non-member distributions. To conclude, the correlation matrix reveals a near-perfect linear relationship between the majority of the risk metrics, with coefficients consistently reaching 1.00, except for a slight deviation in the ML Inference metric (approx. 0.93).

Overfitting risk model

Again, the results showed will be regarding the Adult Dataset only (see Figure 4.3).

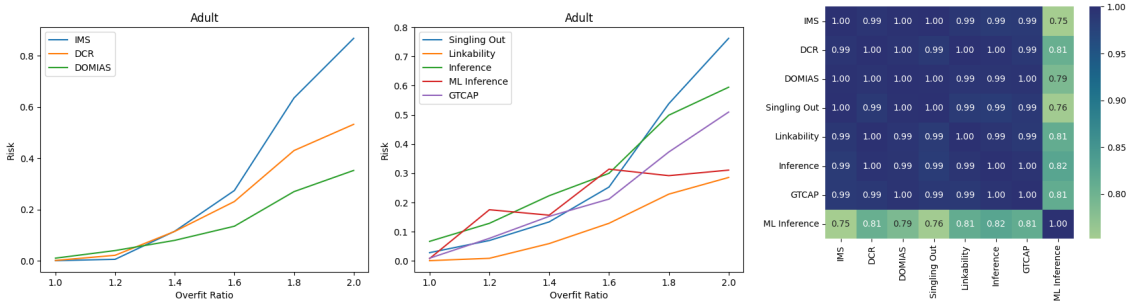


Figure 4.3: Metric comparison results on Adult dataset in the overfit risk model (RTF [122]) and correlation between the metrics

Under the overfitting risk model, the various privacy metrics show a consistent pattern of response to risk, resulting in large correlations between them. Despite employing the maximum amount of risk insertion during the experiments, the resulting measured risks were only about half as severe as those observed in the more theoretical worst-case scenario of the leaky risk model. While correlations between the metrics remain strong, they are weaker compared to the leaky risk model, especially when applied to the Texas dataset. This direct relationship between the degree of generator overfitting and the detected privacy risk emphasizes a key finding: overfitting inherently introduces vulnerabilities in synthetic data generation. This insight can be applied proactively, suggesting the use of early stopping techniques during the generator's training process to enhance privacy protection.

Differential Privacy risk model

Under the differential privacy risk model (see Figure 4.4), the metrics demonstrated negligible sensitivity to increases in the privacy budget. Theoretically, a larger budget implies looser privacy constraints and should result in detectable increases in risk; however, the observed risk scores remained stagnant. This lack of variation renders the calculated correlations uninterpretable, as there is insufficient signal to measure a relationship. We attribute this phenomenon to the poor utility of the DP generators' output—essentially,

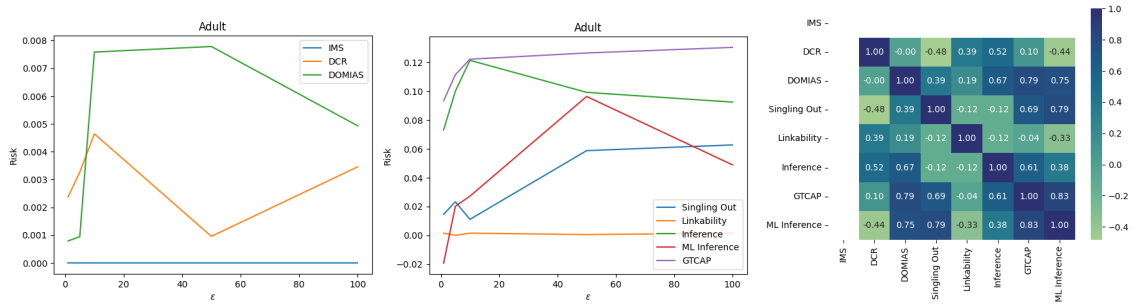


Figure 4.4: Metric comparison results on Adult dataset in the differential privacy risk model (AIM [89]) and correlation between the metrics

the synthetic data was too noisy to retain meaningful structure regardless of the budget setting (see Section 4.5 for the utility analysis).

Outlier removal

Again, in the outlier removal experiment, most of the times the metrics exhibit the same behavior. More on the discussion of the result can be found in the Appendix A.3.

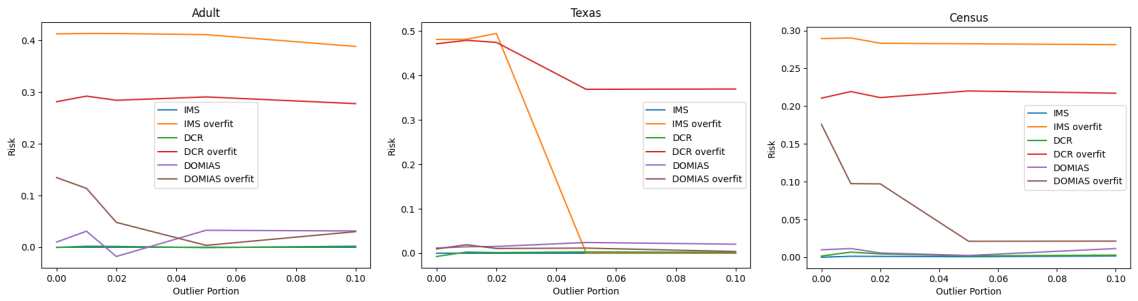


Figure 4.5: IMS, DCR, and MIA with outlier removal in the original dataset prior to generator training, with both no overfitting ($f_o = 1$) and overfitting ($f_o = 1.6$) - RTF [122]

In particular, all the metrics seems to be not susceptible by outlier removal. Most of the times this is caused by the fact that the leak is too low to see an actual difference. We then inspect the case of the overfitted model, in this case is clear how DOMIAS, especially in Adult and Census datasets, is vulnerable to this particular setting (see DOMIAS overfit line). For what regards the reported decrease in the Texas dataset of both IMS and DCR, our findings suggest that the specific properties of a dataset significantly influence how effectively outlier removal can mitigate privacy risks. For instance, the Texas dataset presents a unique challenge because it is heavily categorical (21 of 28 attributes) and many of its attributes possess only a few distinct values. This constrained combination space inherently limits the diversity of possible records, which can make any existing outliers more conspicuous targets and thus vulnerable to a broader array of privacy attacks.

Hyperparameter Sensitivity

To comprehensively evaluate the privacy risk across each dataset and under both the leaky risk model and the overfitting risk model, we systematically experimented with k-Nearest Neighbor (k-NN) based privacy indicators for a range of k values. Our results revealed a clear and consistent finding: the indicator based on $k = 1$ consistently exhibited superior performance and dominance over all other values of k across both risk models. This observed dominance of $k = 1$ is significant because it suggests that the most effective way to detect the specific types of privacy risks induced by these models (direct data leaks or localized memorization due to overfitting) is by examining the immediate neighborhood of a data point. Larger values of k introduce a degree of smoothing or averaging over a broader region of the data space, which dilutes the signal of a very specific, localized leak or memorization event, thereby making the indicator less effective.

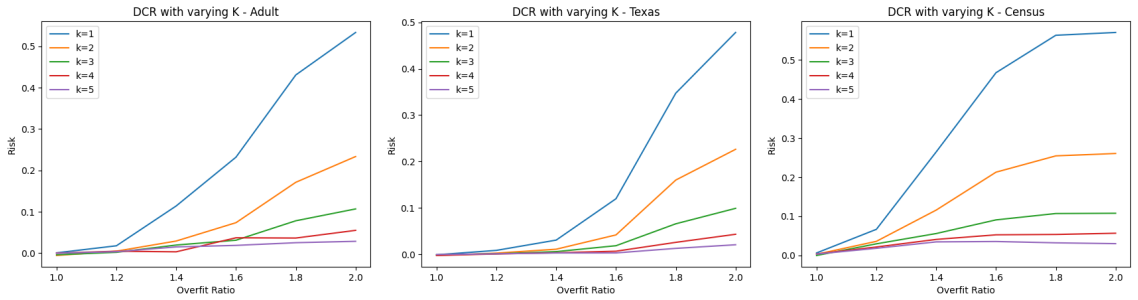


Figure 4.6: Results with k-NN-based privacy metric for various k using the overfit risk model - RTF [122]

We analyzed the impact of radii on GTCAP to determine its effect on the risk measure within the training set, explicitly excluding the control set baseline. For this evaluation, we selected specific attribute subsets across the three domains: for the Adult dataset, we used "income" as the target with "workclass", "education", and "marital status" as keys; for Texas, the target was "length of stay" with "illness severity", "pat country", and "sex" code as keys; and for Census, we selected "incwage" as the target alongside "nchild", "race", and "sex". The results demonstrate that increasing the radii results in higher risk scores, as larger radii broaden the range of values considered equivalent, thereby increasing the potential for overlap.

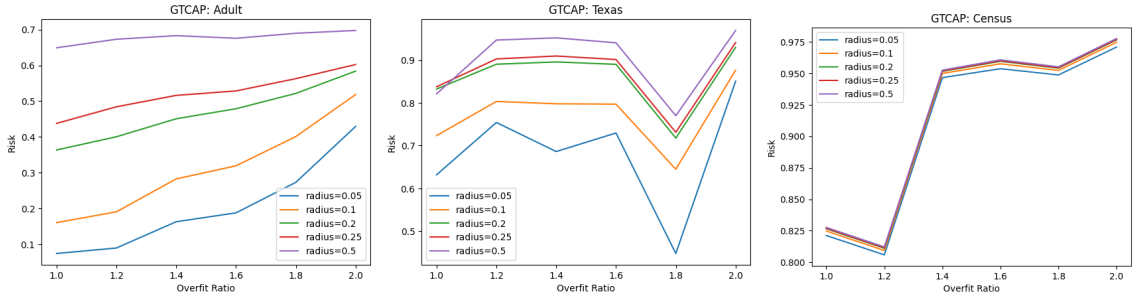


Figure 4.7: Results with GTCAP privacy metric for various radiuses using the overfit risk model

Synthetic data size

In this experiment, the x-axis represents the multiplicative factor applied to the training set size to generate the synthetic data (for example, 4 means $N_{\text{synth}} = 4N_{\text{train}}$). The plot (see Figure 4.8) reveals that the Distance to Closest Record (DCR) metric is uniquely susceptible to this setup, showing a sharp increase in privacy risk as the synthetic dataset grows, while other metrics like GTCAP and Linkability remain stable. This susceptibility arises because DCR relies on raw Euclidean distances; as the number of synthetic points increases, the density of the space grows, naturally reducing the distance to the nearest real record. Consequently, DCR fails to provide a robust privacy assessment in this context without a correction term to account for the disproportionate size of the synthetic dataset relative to the training data.

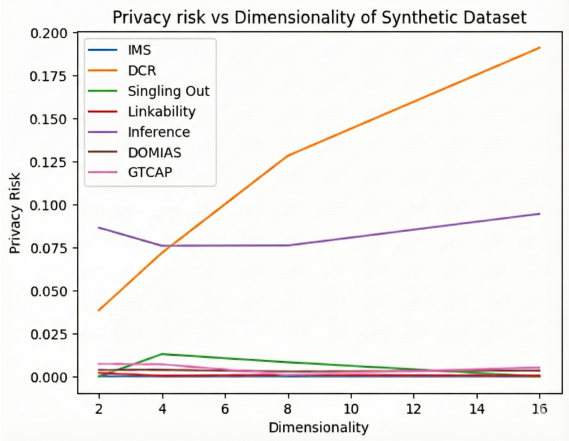


Figure 4.8: Evolution of privacy risk scores as the volume of generated synthetic data increases relative to the original training data.

Utility

To conduct a thorough evaluation of the synthetic datasets, we measured their utility using the Machine Learning Efficacy (MLE) metric. This metric determines how well a binary classifier can distinguish between real and synthetic data samples. To compute the MLE, we created balanced training and test sets by pooling an equal number of real and synthetic records. We then trained an XGBoost Classifier on this mixed data to learn how to differentiate the two origins. The resulting classifier accuracy serves as the MLE score. The interpretation of the score is straightforward: an MLE value close to 0.5 indicates that the synthetic data is statistically very similar to the real data, signifying high utility (the classifier struggles to tell them apart). Conversely, a score closer to 1.0 means the classifier can easily distinguish between real and synthetic records, implying low utility. The specific MLE scores for each dataset and generator are presented in Figure 4.9.

The non-differentially private (non-DP) synthesizers generally achieve higher utility scores (i.e., MLE values closer to 0.5) because they are optimized for fidelity without privacy constraints. Conversely, the differentially private (DP) synthesizers typically exhibit lower utility (i.e., MLE values further from 0.5) as they intentionally inject noise

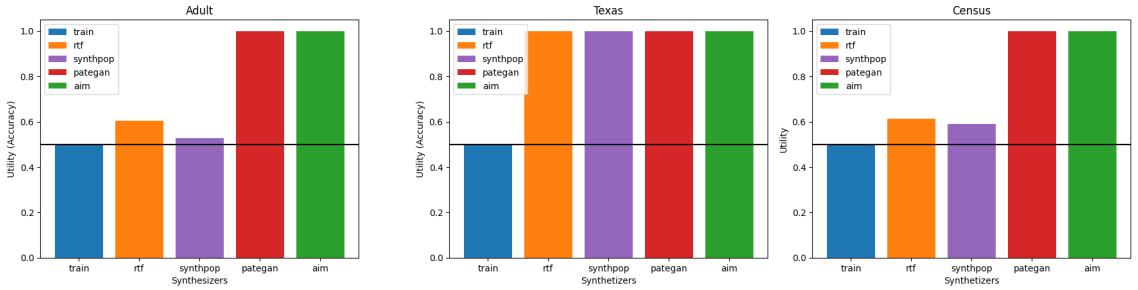


Figure 4.9: MLE utility scores of synthetic datasets per dataset and generator (utility of training set for reference)

or distortion to satisfy their privacy budget. A distinct saturation behavior (score of 1.0) is observed for all the synthesizers on the Texas datasets. This extreme outcome stems from the compounding effects of our strict experimental regime and the intrinsic difficulty of the dataset itself. The Texas dataset is notoriously challenging for generative modeling due to its high dimensionality, sparsity, and high-cardinality categorical features (e.g., thousands of unique diagnosis and procedure codes). Capturing these complex, sparse dependencies requires significant model capacity. However, our setup enforced all regularization measures and restricted DP models to a tight privacy budget of $\varepsilon = 1.0$. Under these constraints, the addition of DP noise effectively drowned out the signal required to learn the dataset's sparse structure. Consequently, the models likely collapsed to a degenerate or uniform distribution that artificially maximized the metric, rather than learning the true underlying distribution.

Canary record baseline

In this section, we evaluate Attribute Inference Attacks (AIAs) on the *Adult* dataset by comparing the canary record baseline [4, 62] against the training set score (details in Section 4.3.4). To ensure robustness, we repeated the baseline computations for 100 randomly sampled target records. The average success rates for standard inference, ML-based inference, and GTCAP are reported below.

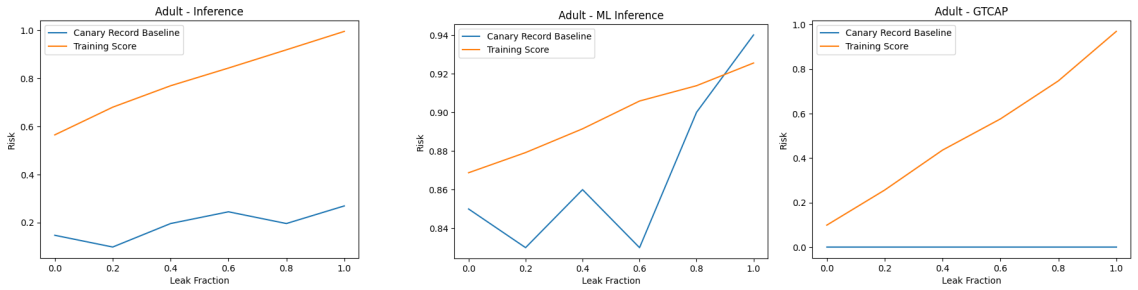


Figure 4.10: Comparison between Canary Record Baseline and Risk on the training set using the leaky risk model)

The Canary Record Baseline framework aligns more effectively with contemporary privacy auditing standards, particularly for synthetic data where traditional generalization gaps often serve as insufficient proxies for privacy risk. This methodology facilitates a

more rigorous and granular assessment of vulnerability, specifically within adversarial scenarios involving active information extraction.

4.6 Conclusions

The experimental findings demonstrate that outliers constitute significant privacy hazards, a threat potentially stemming from localized overfitting around these unique data points. This phenomenon appears to disproportionately affect Membership Inference Attacks (MIAs), which logically focus on identifying and targeting unique individuals. This observation is consistent with established literature concerning MIAs. In contrast, the DCR exhibited a robust ability to detect risks even after outlier removal, consistently showing higher values for overfit models compared to non-overfit models, irrespective of the fraction of outliers removed. Furthermore, the degree to which outlier removal mitigates privacy risks is influenced by the dataset's properties. Datasets rich in categorical attributes with limited value combinations, like the Texas dataset, may render outliers more conspicuous targets for a broader spectrum of attacks. Conversely, datasets with a lower proportion of categorical attributes and wider value ranges, such as the Adult and Census datasets, may restrict the exploitability of outliers to highly specialized methods, exemplified by the DOMIAS attack.

With our work we present a comprehensive framework designed for the empirical assessment of the efficacy of tabular synthetic data privacy metrics. To establish a foundation for this assessment, the work first surveys existing metrics, categorizing them into: mathematical privacy properties; statistical privacy indicators; and simulated attacks. The simulated attacks are further classified into Singling Out, Linkability, and Inference attacks, establishing a direct correlation with pertinent legal theories on anonymization. The core of the framework involves evaluating metrics across three novel risk models specifically engineered to enable the empirical measurement of metric responsiveness to deliberately introduced risk. These models systematically insert risk through: direct data leaks; generator overfit; and the manipulation of privacy budgets for Differential Privacy (DP) generators. Furthermore, the framework employs baseline computations to address the discrepancy between two distinct types of information inferred by generative models: specific information pertaining to the generator's training data, and general information concerning the underlying population. Experimental application of the framework revealed a strong correlation among no-box privacy quantification methods, suggesting that uniqueness-based and similarity-based risks largely coincide when evaluated under no-box risk models. This finding implies that the selection of a privacy quantification method might be optimally guided by considerations of robustness and efficiency, a criterion that would favor statistical indicators. However, it is noted that such indicators explicitly focus only on the distances between real and synthetic data points. Consequently, future research should explore the feasibility of reliably, robustly, and efficiently quantifying uniqueness as a distinct privacy risk factor for synthetic data at the level of probability distributions, following the direction set by Acquisto et al. [36].

Chapter 5

Hybrid pipeline for Synthetic Data Generation

This chapter presents a systematic empirical evaluation of the proposed hybrid pipeline, which integrates classical anonymization techniques with modern generative models to enhance data privacy. By transitioning from theoretical architectures to practical experimentation, we aim to quantify how this synergetic approach affects the trade-off between statistical utility and disclosure risk across diverse data environments. The main contributions are:

- **Hybrid Data Synthesis Pipeline**: We design a novel, multi-stage architecture that integrates k-anonymity and stratified sampling as pre-processing steps for generative models, ensuring a "privacy-by-design" guarantee that purely stochastic models cannot provide.
- **Taxonomy of Transition Steps**: We define and formalize three mechanisms—Single, Uniform, and Conditional—for transitioning data from generalized equivalence classes back to fully specified synthetic inputs.
- **Identification of the Transition Paradox**: We provide empirical evidence of a "transition paradox", where high-fidelity transition methods (like Conditional sampling) inadvertently re-introduce the specific privacy risks that anonymization sought to remove.
- **Elimination of Residual Risk**: We demonstrate that while k-anonymity alone often leaves residual Singling Out risks, the subsequent application of the generative model in our pipeline successfully reduces this risk to near-zero, validating the efficacy of the layered defense strategy.

5.1 Motivations

Data anonymization and synthetic data generation through AI represent two distinct but often complementary approaches to privacy-preserving data utility, each possessing a unique set of strengths and weaknesses that will be examined in the following sections.

5.1.1 Data Anonymization

Data anonymization has been extensively presented in Section 2.2.1. Traditional Data Anonymization creates a sanitized version of a dataset by employing techniques such as generalization and suppression to modify identifying values. Its primary objective is to satisfy syntactic privacy models like k-anonymity, l-diversity, and t-closeness, which ensure that individuals are indistinguishable within a group or that sensitive attributes are well-distributed [86, 126]. Its primary strength lies in its ability to preserve the truthfulness of the underlying records, as the data remains a modified version of the original ground truth rather than a simulation. However, these methods suffer from severe limitations when applied to high-dimensional or complex datasets. The necessary suppression and generalization of data to satisfy privacy constraints often result in a catastrophic loss of data utility, destroying the fine-grained correlations required for machine learning tasks [16]. Furthermore, traditional anonymization is notoriously vulnerable to linkage attacks; if an adversary possesses an auxiliary dataset (e.g., a voter registration list), they can often cross-reference "anonymized" quasi-identifiers (like zip code and birth date) to re-identify individuals with high confidence [94, 126]. This "curse of dimensionality" means that as a dataset becomes more detailed and useful, it becomes exponentially harder to anonymize effectively without rendering it statistically useless. Another important issue with data anonymization relates to the format of the sanitized dataset.
As we have already seen in Section 2.2.1, the anonymized dataset differs from the original one in terms of data types and attribute names. In particular, through the generalization process, numeric quasi-identifiers are binned, transforming them into categorical variables from a practical point of view. Instead, categorical quasi-identifiers are usually generalized by replacing the original attribute with a more generic one. These transformations break the original structure of the table, rendering the output incompatible with downstream systems that rely on strict schema definitions. This structural divergence creates a significant operational bottleneck, particularly in production environments where data processing pipelines and legacy software are engineered to expect specific data types (e.g., continuous integers) rather than string-based generalizations. Consequently, utilizing such anonymized datasets often necessitates expensive software refactoring or the complete retraining of machine learning models to accommodate the altered feature space, thereby introducing a "technical debt" that can outweigh the benefits of data access.

5.1.2 Synthetic Data Generation

Synthetic Data Generation (SDG) represents a paradigm shift from "sanitizing" real data to "simulating" it, as previously discussed in Section 2.2.2. By employing complex generative architectures such as Generative Adversarial Networks (GANs), Synthpop or, more recently, Diffusion Models, SDG aims to approximate the joint probability distribution $\mathbb{P}_{\text{data}}$ of the original records to sample entirely new, artificial data points.
A primary theoretical motivation for adopting this approach is the potential to decouple released information from the original data subjects. In this framework, the absence of a direct bijective mapping between real individuals and synthetic records suggests a conceptual shift that might reduce the risk of direct re-identification compared to tradi-

tional anonymization. This capability allows for the preservation of complex, non-linear relationships and multivariate correlations that are typically destroyed by the coarse generalization or cell suppression inherent in traditional anonymization.
However, SDG is not a silver bullet and faces significant scrutiny regarding its privacy claims. The major disadvantage is the lack of formal, verifiable privacy bounds in standard generative models. Deep generative models are inherently designed to minimize the divergence between real and synthetic distributions; in doing so, high-capacity models like GANs can overfit and inadvertently memorize outliers or unique sequences from the training data, reproducing them verbatim in the synthetic output. This vulnerability exposes the data to Membership Inference Attacks (MIA), where an attacker can determine probabilistically whether a specific individual's data was used to train the model, effectively undoing the anonymity promise. Furthermore, without the mathematical guarantees provided by frameworks like Differential Privacy, the safety of synthetic data remains purely empirical. It must be validated through extensive post-hoc privacy attacks. This creates a black-box of privacy assurance that is computationally expensive to audit and difficult to certify for strict regulatory compliance.

5.1.3 The Rationale for Hybridization

The complementary nature of the limitations described in the previous sections suggests that neither paradigm is sufficient in isolation for high-stakes scenarios requiring both rigorous privacy and usable data utility. This necessitates a hybrid approach that integrates the strengths of both methodologies. By applying data anonymization techniques (achieving k-anonymity for example) to the source data prior to the training of generative models, one can effectively "sanitize" the training distribution. This pre-processing step serves as a privacy firewall, ensuring that the generative model learns from a distribution where sensitive outliers have already been suppressed or generalized, thereby mitigating the risk of the model memorizing and leaking unique individual traits [124]. Consequently, the resulting synthetic dataset inherits the formal privacy guarantees of the anonymized input while leveraging the generative model's capacity to reconstruct complex statistical structures, potentially smoothing out the utility distortions introduced by strict anonymization.

5.2 Proposed Methodology: The Hybrid Data Synthesis Pipeline

To the best of our knowledge, this is the first formal work to combine these two approaches in a sequential pipeline. The entire three-stage pipeline is illustrated conceptually in Figure 5.1. The pipeline is designed to enforce privacy a priori while mitigating the resulting utility loss. The process begins by dividing the original dataset into two non-overlapping sets: the training set and the test set. This separation is crucial for the objective evaluation of both utility and privacy metrics post-synthesis.

The hybrid pipeline consists of three fundamental sequential stages:

1. **Data Anonymization** (Privacy Enforcement)

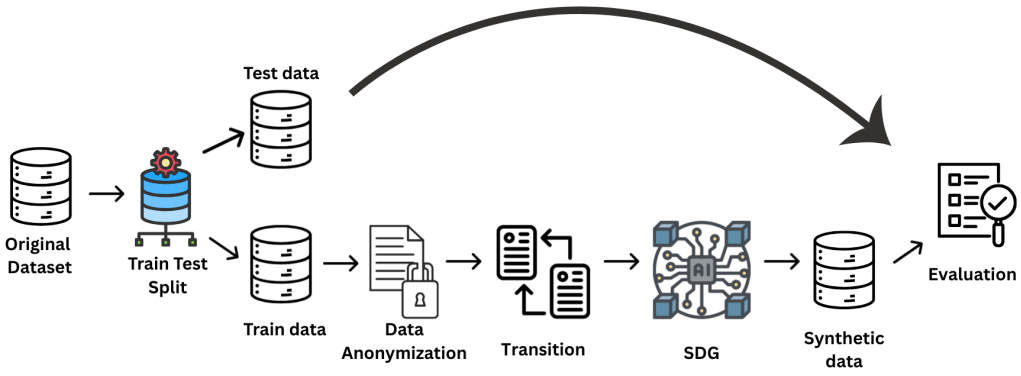


Figure 5.1: Hybrid pipeline: combining data anonymization with synthetic data generation.

2. **Transition** (Utility Reconstruction)

3. **Synthetic Data Generation (SDG)** (Statistical Modeling)

5.2.1 Data anonymization Step

Our data processing pipeline initiates with a foundational and critical step: data anonymization. This phase utilizes a blend of two complementary techniques, i.e., generalization and record suppression, to rigorously enforce a predefined privacy model. For our experimental validation, we concentrate on the principles of k-anonymity and l-diversity. We plan to test k-anonymity across a pragmatic range of values, specifically $k \in \{5, 10, 20, 50\}$, while holding the l-diversity parameter constant at $l = 2$. This selection allows for a systematic evaluation of the privacy-utility trade-off, ranging from the standard minimum protection threshold of $k = 5$ to more conservative configurations , while holding $l = 2$ ensures a baseline defense against attribute disclosure by requiring at least two distinct sensitive values within each equivalence class. Subsequently, we will expand this analysis by testing higher values of l to investigate how increasing the variety of sensitive values within each equivalence class further mitigates inference risks at the potential cost of reduced data utility. The implementation begins by systematically identifying the Quasi-Identifiers within each dataset, which are the attributes an attacker could use for linkage. Once identified, we construct generalization hierarchies that map specific data points to broader, less identifying categories. These hierarchies are then supplied to the ARX software, a sophisticated anonymization tool. ARX employs a complex optimization algorithm designed to compute the most effective combination of generalization transformations that satisfies the required privacy level (e.g., $k = 5$)

while simultaneously ensuring the least possible loss of data utility.

Generalization is the core utility-preserving mechanism, but it has limitations. Certain records may be inherently unique, meaning they cannot be placed into an equivalence class without demanding an excessively coarse level of generalization that would negatively impact the utility of the entire dataset. This is the precise scenario where record suppression becomes indispensable. Suppression acts as a safety valve, removing those specific outlier records to guarantee that the privacy constraints are met without sacrificing the overall data quality of the remaining records. We have set a highly permissive suppression limit of 100%, meaning ARX is authorized to suppress any record it cannot fit optimally, allowing it to prioritize the integrity of the generalization solution. However, we have observed a practical constraint: when dealing with datasets that contain a large number of QIs, the computational complexity of the optimization problem often becomes intractable, sometimes resulting in an overflow error and preventing the software from finding a viable anonymization solution [92].

Once the dataset satisfies the constraints imposed by the chosen anonymization model, the following step of our pipeline applies a sampling mechanism. While sampling alone does not constitute an anonymization technique, since it does not enforce properties such as k-anonymity, it introduces uncertainty regarding an individual's inclusion in the released dataset. This uncertainty is closely related to the notion of membership privacy, a foundational concept underlying differential privacy.

While plausible deniability [11] has been proposed as a privacy notion for synthetic data generation—ensuring that no output record can be confidently attributed to a specific individual in the original dataset—it is conceptually distinct from classical anonymization models such as k-anonymity. In particular, plausible deniability is defined with respect to the data generation mechanism, whereas k-anonymity constrains the structure of the released dataset itself. For this reason, plausible deniability cannot be directly achieved through anonymization alone, nor can it be assumed to emerge implicitly from k-anonymity.

Prior work has shown that the interaction between sampling and anonymization can yield formal privacy guarantees. In particular, Li et al. [81] demonstrate that applying random sampling before a "safe" k-anonymization procedure results in a relaxed differential privacy guarantee, denoted $(\beta, \varepsilon, \delta)$-DP. Conversely, when anonymization precedes randomization—as in our proposed approach—the resulting protection can be interpreted through the lens of group differential privacy, where privacy guarantees scale with the size of the indistinguishable group [23] (e.g., (c, ε, δ)-DP).

Although this framework does not satisfy the formal definition of plausible deniability as introduced for synthetic data generation, it nevertheless establishes a principled connection between classical anonymization techniques and modern probabilistic privacy notions. By combining k-anonymity with sampling, our approach might achieve a theoretically grounded form of privacy protection that strengthens traditional anonymization while remaining compatible with differential privacy–based interpretations.

To correctly formalize the final step of our privacy-preserving framework, it is essential to detail the mechanism of the sampling procedure. Specifically, we employ β-sampling, a technique formally recognized in privacy literature [6]. This refers to a probabilistic

selection process where every single record from the original, anonymized dataset is independently included in the subsequent processing set (often a training set) with a fixed probability β, where the probability must be greater than zero and less than or equal to one ($0 < \beta \leq 1$). This inherent stochasticity is the mathematical element that significantly amplifies the overall privacy guarantees of the pipeline. In simple terms, the lower the value of β, the higher the resultant uncertainty for any malicious adversary attempting to definitively verify an individual's presence in the final output.
While **Simple Random Sampling**, the method of selecting a specific number of records, n, with equal probability, is the most straightforward realization of this probabilistic concept, the selection of an appropriate sampling methodology is critically dependent on the structure and composition of the data. To ensure comprehensive representation and maintain statistical integrity, **Stratified Sampling** offers particular value within this hybrid privacy pipeline. This method systematically partitions the dataset's population into homogeneous subgroups, known as strata, before the independent β-sampling occurs within each group. This targeted approach is vital because it prevents specific minority groups, or the integrity of specific k-anonymity equivalence classes, from being accidentally eliminated or underrepresented during the down-sampling process. By ensuring proportionate representation across all sensitive groups, stratified sampling actively preserves the statistical utility and fairness of the data, which is crucial for the subsequent stage of synthetic data generation.
For the empirical phase of our experiments, we will conduct tests using both simple random sampling and stratified sampling. The β parameter will be rigorously selected from a standard set of values: $\{0.1, 0.2, 0.5, 1.0\}$. These particular values are commonly utilized within privacy and machine learning research to systematically check and quantify the trade-off between the level of privacy achieved and the resulting loss in data utility. Evaluating the framework across this range of β values allows for a robust empirical assessment and a clear demonstration of the privacy amplification achieved by the sampling step.

5.2.2 Transition Step

The Transition Step is arguably the fundamental and most challenging part of this entire privacy pipeline, serving as the crucial bridge between the anonymized state and the final utility-preserving format. The primary purpose of this step is to transform our intermediate, anonymized dataset back into a format that closely mirrors the structure and properties of the original sensitive data. This transformation is driven by a critical user requirement: consumers of the final dataset expect the data attributes to retain the semantic meaning and format of the source data, even if the values themselves are now privacy-enhanced. Essentially, we are tasked with subtly removing the initial anonymization layer (the coarse generalization) to restore data utility without simultaneously reintroducing the original, identifying information. If we were to simply copy the original, specific values back into the dataset, we would entirely nullify the extensive effort and purpose of anonymizing the data in the first place.

The challenge, therefore, lies in reversing the generalization without revealing too much information about any single record. To manage this delicate balance, we have defined a

taxonomy of three distinct types of transition mechanisms: single, uniform, and conditional. The execution of each of these transition types is context-dependent, meaning the procedure is uniquely tailored based on the type of the attribute being processed, specifically, whether the attribute is numeric or categorical. For instance, a generalized numeric range must be re-specified to a single, believable number within that range, whereas a generalized categorical value (like "Italy" for a country) needs to be consistently resolved back to a specific value (like a specific city name) while preserving privacy guarantees achieved by the k-anonymity equivalence class. The careful choice and implementation of these transition types for both numeric and categorical attributes ensures that the resulting data retains maximum utility and is immediately usable by end-users, while still adhering to the stochastic privacy guarantees established in the previous sampling phase.

Single

The first defined method within our transition step is the Single transition. The defining characteristic of this approach is its commitment to identifying and assigning a single, consistent value for a generalized attribute. This identified value is then uniformly applied across all records that share the exact same generalized attribute value. The goal is to collapse a generalized category or range back into a specific, discrete data point.
For numeric attributes, the generalization layer, which typically represents an attribute through a range (e.g., age $[20 - 29]$), is removed by selecting one definitive value from that range for every record. In our experiments, we prioritize central tendency measures to minimize distortion: we specifically utilize the mean or the median of the given range. For instance, if the Age attribute has been generalized into the range $[20 - 29]$, every record containing this range will have the value 25 substituted back in. Analogously, all records initially generalized to the range $[30 - 39]$ will subsequently be replaced with the value 35.
When dealing with categorical attributes, we have two primary methods for executing the Single transition. The first option is to select a specific value from the leaf nodes of the original generalization hierarchy. The second, and often more statistically robust, option is to select the mode, that is, the most frequently occurring representative value, within that generalized category from the original, non-anonymized data. For example, consider the Occupation attribute, which might be generalized by sector into categories like Healthcare or Education. To remove this generalization, we could select a random instance from the original category, replacing Education with Professor and Healthcare with Doctor. Alternatively, using the mode, we might consistently substitute the Healthcare category with Nurse if that was the most common specific occupation in the original data.
While the Single transition successfully restores utility by presenting specific, non-ranged values to the end-user, this method has a distinct drawback: it inherently decreases the variability of the dataset. By repeatedly assigning the exact same specific value to a set of generalized records (e.g., assigning 25 to everyone aged $[20 - 29]$), we introduce a form of noise and repetition into the data. Despite this artificial decrease in variability, the fundamental properties of the underlying privacy model are preserved. In the case of k-anonymity, the transition maintains the privacy guarantee because, even after the single value assignment, we will still have at least k records that share the identical com-

bination of attribute values for the quasi-identifiers, meaning the anonymity set remains intact.

Uniform

The Uniform transition represents a deliberate shift in our strategy compared to the Single transition, as it introduces controlled randomness to restore data utility. Instead of assigning a single, deterministic value to an entire generalized structure, this method involves making a probabilistic selection from within each generalization structure, whether it be a numeric range or a categorical hierarchy.
For numeric attributes, the uniform transition is realized by sampling a random value from a continuous uniform distribution defined by the bounds of the generalized range. Specifically, the value $v(d, A)$ for a specific record d and attribute A is drawn from a uniform distribution over the semi-open interval $[a, b[$, where a and b represent the lower and upper bounds of the generalized range, respectively. Crucially, this sampling procedure is performed individually for every record within an equivalence class. This design choice guarantees that, unlike the Single transition, the sampled values across records within the same equivalence class will be different, thereby significantly improving the overall variability and statistical richness of the resulting dataset. However, this randomness is a double-edged sword: while it enhances variability, the independent nature of the sampling could potentially break the intrinsic correlations that existed between the records and their attributes in the original dataset.
The approach for categorical attributes is analogous. The generalization structure (the hierarchy) defines a set of possible, more specific values. The uniform transition samples a value randomly from this set of possible values defined by the generalized category for that equivalence class. For example, consider an anonymized record with Occupation generalized to Healthcare sector and the non-generalized Education attribute recorded as PhD. By applying the uniform transition, the Healthcare sector could be randomly assigned a low-level occupation (like "Janitor") which is highly improbable given the associated PhD education level. This scenario demonstrates how the random nature of the uniform transition can inadvertently disrupt the internal consistency and correlations between attributes within the same record.
It must be acknowledged that due to this inherent randomness, the Uniform transition carries a risk: we might lose some of the specific privacy model properties that were injected earlier in the pipeline, particularly the perfect structural integrity of the k-anonymity equivalence classes. However, this is not a complete loss of protection. We can still confidently make strong considerations regarding the ongoing privacy protection due to the significant uncertainty added throughout the pipeline—stemming from both the initial anonymization step and the stochastic way the transition procedure is performed. The combination of generalization, suppression, β-sampling, and now uniform random assignment provides a high degree of plausible deniability by ensuring that no single output value can be definitively traced back to a specific original record.

Conditional

The final and most complex mechanism within our framework is the Conditional transition. This transition type is specifically designed to inject the maximum amount of

information back into the dataset while still maintaining the privacy achieved earlier in the pipeline. It directly addresses the primary weakness of the Uniform transition, which is the tendency to break inherent correlations between attributes.

To achieve this high-fidelity transition, we adopt a procedure where, for every equivalence class generated during the anonymization step, we internally store the empirical conditional distribution of the quasi-identifiers. This is accomplished by grouping the records belonging to the equivalence class and simply counting the instances of each unique attribute value within that group. By sampling from this stored, informative distribution, which is maintained securely within the pipeline and is not made available to the end-user, we effectively avoid the problem of breaking correlation between records and attributes. The samples generated are statistically realistic because they reflect the actual frequencies present in the original data subset that formed the equivalence class. For a concrete example, consider an Age attribute generalized to the range $[20 - 29[$ within an equivalence class composed of five original values: $\{21, 21, 23, 28, 22\}$. When performing the conditional transition, the sampling process for this range will assign a value of 21 with a probability $p = \frac{2}{5}$ (or 40%), while the other values (23, 28, 22) will each have a lower probability of $p = \frac{1}{5}$ (or 20%).

This approach offers significant potential for refinement. The representation of the conditional distribution can be further improved by utilizing advanced density estimation methods, such as the Kernel Density Estimator (KDE). Employing KDE can provide a smoother, more generalized representation of the underlying distribution from which to sample values, leading to a more realistic outcome than simple histogram counting. Furthermore, to generate an even more realistic sample, this method can be extended to take into consideration the values of the other attributes within the equivalence class, thereby estimating a multivariate probability function that captures the true interdependence between QIs.

However, the Conditional transition comes with two major, intertwined downsides that necessitate careful risk assessment. First, by incorporating such rich distributional information back into the data, we significantly increase the probability of injecting too much information, consequently elevating the risk of leaking private information about specific individuals, even if the equivalence class structure remains technically intact. Second, the introduction of sophisticated machine learning models, like KDE or other techniques needed to estimate multivariate probability functions across potentially hundreds of equivalence classes, leads to a substantial increase in the computational resources required to execute the entire pipeline, impacting its scalability and performance.

5.2.3 Synthetic Data Generation Step

The final and conclusive step in our pipeline is the Synthetic Data Generation (SDG) phase. At this stage, we train a synthetic data generator on the dataset that has undergone anonymization, sampling, and the chosen transition step. The core principle here is that the generator is trained on a dataset that, by design, no longer contains the specific, highly identifying private information of any single individual. Consequently, the resulting synthetic dataset should be fundamentally incapable of perfectly replicating that sensitive information, as the generator never had access to it in its original form.

We can illustrate the purpose of this entire framework using the analogy of a painter

attempting to replicate a portrait. The original dataset is the portrait, the synthetic data generator is the painter, and the specific private features (like unique scars or very specific birthmarks) are the highly identifying details. A key risk in standard SDG is overfitting, that is, where the "painter" reproduces specific, unique points, such as outliers. By performing the preceding data anonymization and sampling steps, we are effectively blurring the original portrait before handing it to the painter. The painter can no longer see those specific, highly unique features because they have been hidden, generalized, or suppressed during the initial process. Therefore, the generator cannot reproduce them in the synthetic output, achieving a high degree of privacy protection.
For the implementation of this phase, we utilize three distinct types of state-of-the-art synthesizers, models that leverage different Artificial Intelligence paradigms to learn the complex multivariate distribution of the data. The models we are using are CTGAN [137], Tabddpm [72], and REalTabFormer [122]. These methods, which have been previously detailed in Section 2.2.2, vary significantly in terms of the underlying AI models they employ, that is, a Generative Adversarial Network for CTGAN, a Diffusion Model for Tabddpm, and a Transformer-based model for REalTabFormer. This difference leads to varied computational requirements, time complexity, and configuration needs.
The training regimens for each model are carefully controlled to ensure a robust evaluation.

- CTGAN is trained for 300 epochs, utilizing a learning rate of 10^{-4} consistently applied to both its generator and discriminator components.
- Tabddpm, a diffusion model, requires more iterations for convergence and is consequently trained for 1000 iterations. It also uses a learning rate of 10^{-4} and employs three hidden layers, each containing 256 neurons, with a dropout probability set to 0.1 as a regularization technique.
- REalTabFormer, a Transformer-based architecture, is trained for 50 epochs. We use a training-validation split of 90% for the training set and 10% for the validation set. For regularization, we employ both early stopping and dropout, utilizing their respective default parameter values to prevent overfitting and ensure the model generalizes well to unseen data distributions.

This final step completes our privacy-preserving pipeline, resulting in a synthetic dataset that is statistically representative, high in utility, and fortified by layered privacy guarantees against re-identification.

5.2.4 Analysis of Pipeline Order and Flexibility

The current implementation of our pipeline is highly flexible, allowing us to perform the individual steps, i.e., data anonymization, sampling, transition, and synthetic data generation, in various orders or even to execute them in isolation. This modularity means we can obtain results using only SDG or only data anonymization if the experimental design requires it.
A core consideration regarding the order lies in the placement of the Transition step relative to the Synthetic Data Generation (SDG) step. Our pipeline design allows for the

Transition to be performed either before (PRE-transition) or after (POST-transition) the SDG process, which is controlled by a specific flag parameter.
We believe that conducting data anonymization after the synthetic data generation would compromise the integrity of our privacy considerations. The SDG process, while reliable, can subtly alter or destroy some intrinsic statistical properties of the dataset. Applying a strict privacy model like k-anonymity post-hoc could lead to a corrupted or unreliable privacy outcome because the underlying data structure has been artificially perturbed. For the Transition step, however, a discussion on its placement only makes sense after the data anonymization has been completed, as its purpose is to reverse the generalization performed in that initial step.
We have empirically tested both the PRE-transition and POST-transition configurations and observed notable differences in performance and privacy leakage.
One significant observation concerns the performance of Tabddpm, the diffusion-based synthesizer, particularly when dealing with categorical data. When we perform POST-transition, the dataset still contains numerous categorical attributes due to the preceding anonymization techniques. This forces the Tabddpm model to heavily rely on computationally intensive processes like One-Hot Encoding and complex Multinomial Diffusion during training. Consequently, the POST-transition order leads to a marked increase in the elapsed time for computations when using Tabddpm compared to other models.
More critically, the choice of order has a direct impact on privacy leakage, particularly when using the Conditional transition. As noted, the Conditional transition samples from the original attribute distribution, meaning it can sometimes sample the original, un-anonymized value for certain individuals.

- In the POST-transition scenario, where the transition happens after SDG, there is a risk that the original value, when sampled, could leak the full original row for some individuals, as the transition directly modifies the final output.
- Conversely, if we perform the PRE-transition and then feed that slightly de-generalized data into the SDG phase, the SDG process itself acts as a further protective layer. The inherent randomness and distributional learning within the data generation phase can effectively mask or cover this behavior, preventing the full row from being deterministically leaked, thereby bolstering the privacy outcome.

Therefore, although the pipeline is flexible, placing the Transition step before the Synthetic Data Generation step is generally preferable for mitigating the risk of inadvertent privacy leakage when using the high-fidelity Conditional transition.

Beyond the execution order, a crucial aspect of the pipeline's flexibility lies in its architectural modularity. While this thesis experimentally validates the framework using k-anonymity and specific deep generative models (CTGAN, TabDDPM, REaLTabFormer), the pipeline is fundamentally model-agnostic.
Practitioners are free to substitute the anonymization module with other privacy enhancing technologies (e.g., t-closeness or other anonnymization techniques) or swap the generative model for a different architecture (e.g., a Bayesian network), as long as the Transition Step is correctly treated. This step acts as a universal adapter: its primary responsibility is to resolve the specific constraints introduced by the chosen anonymization technique (such as generalized ranges or suppressed values) into a valid numerical

or categorical input format that the subsequent generative model can process. As long as this 'bridge' is maintained, the specific algorithms on either end can be interchanged to suit different domain requirements.

Summary of Theoretical Claims and Experimental Validation.

The hybrid architecture detailed in Sections 5.1 and 5.2 relies on two primary theoretical claims. First, we posit that integrating k-anonymity as a structural pre-processing step creates a "privacy-by-design" bottleneck that stochastic Deep Generative Models cannot guarantee on their own, effectively neutralizing the risk of Singling Out by ensuring indistinguishability within equivalence classes. Second, we identify a "Transition Paradox", hypothesizing that while conditional sampling strategies (used to reverse the anonymization) maximize utility, they inevitably re-introduce specific privacy vulnerabilities that the initial suppression sought to eliminate.
In the subsequent experimental evaluation, we empirically validate these claims by benchmarking the hybrid pipeline against standard generative baselines. We systematically vary the Transition Mechanisms (Single, Uniform, Conditional) and k-anonymity thresholds to quantify the precise trade-off between the reduction in Singling Out risk and the preservation of downstream Machine Learning efficacy.

5.3 Evaluation and Preliminary Results

For the empirical validation of our complete pipeline, we have selected three distinct datasets, though the preliminary results presented herein focus exclusively on the widely-used Adult dataset [7]. The Adult dataset allows for a strong baseline comparison with existing literature. In subsequent phases of this research, we intend to expand our evaluation by testing additional datasets to further assess the generalizability and robustness of the proposed framework across different data distributions and dimensions.

In defining our evaluation methodology, we must carefully distinguish between three key measurement categories: privacy, utility, and fidelity.

Selection of the Privacy Metrics

For the privacy metrics, our selection is guided by the Anonymeter framework [49]. This choice is rooted in several technical and regulatory considerations. Firstly, traditional similarity metrics are often difficult to interpret, especially when needing to account specifically for the re-identification risks outlined in the stringent WP29 [5]. Furthermore, recent research, notably in [138], has demonstrated that the standard DCR is suboptimal for reliably quantifying specific privacy risks such as uniqueness or membership. Instead, those authors suggest utilizing MIAs for a more robust interpretation of privacy loss.
However, in our preliminary results, we have made the decision not to use complex MIAs, such as shadow modeling approach, due to practical computational constraints. Additionally, the specific nature of our pipeline's first step—the removal of outliers via the record suppression phase—makes certain advanced MIA variants unsuitable [91, 124,

138]. As was explored in Chapter 4, the alternative to shadow modeling is the DOMIAS attack, but this technique is known to highly exploit the presence of outliers in the training set, meaning it would not be a fair or suitable measure for data that has already been scrubbed of outliers by our pipeline.
Finally, we also excluded the contrastive learning-based framework developed in Chapter 3 from this specific evaluation. As detailed in Section 3.4.3, that framework incurs a significantly higher computational overhead compared to standard metrics, making it less feasible for the extensive, multi-configuration experiments required here. Furthermore, since the core contribution of the Chapter 3 framework is its enhanced ability to detect density-based outliers (via LOF), its application is less critical in this context where the pipeline's design explicitly suppresses such outliers during the pre-processing phase. Therefore, we prioritized the standard Anonymeter suite to maintain a balance between evaluation depth and computational efficiency.

Utility and Fidelity Metrics

For utility metrics, we employ Machine Learning Efficacy. This involves training a standard classifier model, such as XGBoost, to correctly distinguish between real and synthetic records. Poor performance suggests that the synthetic data has lost crucial statistical relationships necessary for effective model training, signaling low utility.

Finally, for fidelity metrics, which measure how closely the synthetic data distribution matches the original data distribution, we use the Wasserstein distance and the Total Variation Distance (TVD). The Wasserstein distance (or Earth Mover's Distance) is dedicated to evaluating the fidelity of numeric attributes. This metric computes the minimum "cost" required to transform one probability distribution into another, effectively comparing the synthetic distribution with the original distribution of a numeric attribute. This is formalized as:

$$W(P, Q) = \int_{-\infty}^{+\infty} |F_P(x) - F_Q(x)| \, dx \tag{5.1}$$

where P and Q are the two probability distributions being compared, and $F_P(x)$ and $F_Q(x)$ are their respective cumulative distribution functions (CDFs).
For categorical attributes, we utilize the Total Variation Distance (TVD). This metric compares the discrete probability distributions of the two attribute columns and is defined by the following equation:

$$TVD(P, Q) = \sum_{x \in \mathcal{X}} |P(x) - Q(x)| \tag{5.2}$$

where, $\mathcal{X}$ is the set of all possible categories, and $P(x)$ and $Q(x)$ are the probability masses (relative frequencies) of category x in the two distributions being tested. Informally, the TVD quantifies the maximum information gain an adversary could achieve if they used the synthetic distribution P as an approximation of the true distribution Q.

Results

We will now examine the results presented in Figure 5.2, which visualizes the normalized scores for our privacy, utility, and fidelity metrics across three key moments in the pipeline. The "Original" stage establishes a baseline using the training data itself; the "Anonymized" stage shows results after data anonymization and the transition step; and the "SDG" stage represents the final outcome after synthetic data generation. We normalized the values using the Min-Max scaler to clearly distinguish between the metric types.

For the utility and fidelity metrics (ML Efficacy, Wasserstein Distance, and TVD), we set the scale such that a value of 0.0 signifies perfect utility or fidelity. These metrics correctly begin at or near 0.0 in the "Original" setup. As the pipeline progresses, their values increase, which is the expected consequence of utility decreasing. This decrease occurs because the anonymization step intentionally injects noise and corrupts the data through generalization. A crucial finding is that the gap between the "Original" and "Anonymized" scores is significantly larger than the gap between the "Anonymized" and "SDG" scores. This indicates that the anonymization phase accounts for the vast majority of the utility loss, with the synthesizer learning to reproduce the already corrupted dataset without adding much further degradation.

Conversely, for the privacy metrics, that is, Singling Out risk, Linkability, and Inference, a score of 1.0 represents a total privacy leak, and 0.0 represents perfect privacy. In the "Original" setup, the SO Risk and Inference scores correctly begin at 1.0, confirming maximum information leakage. After the data anonymization step, all privacy metrics show a dramatic improvement: Inference risk immediately drops to 0.0, indicating complete success in obscuring the ability to infer sensitive attributes. The Singling Out risk decreases substantially to approximately 0.3 at the "Anonymized" stage, and only reaches its minimum, 0.0, after the final "SDG" phase. This demonstrates that the synthetic data generation step provides the necessary final layer of uncertainty to fully eliminate the risk of singling out an individual. Notably, the Linkability metric is not visible on the plot because its score was 0.0 at every step of the pipeline. This result is currently under investigation, as it suggests that considering only one neighbor for the linkability attack is insufficient to generate a measurable risk signal.

Focusing on the trends exhibited in the initial metrics plot, rather than the specific normalized values, we can discern the core functional purpose of each pipeline stage. A particularly interesting trend is observed in the Singling Out risk. Even after the rigorous anonymization process (the "Anonymized" stage), the SO Risk still shows a small, nonzero signal (approximately 0.3). This observation strongly suggests that the initial data anonymization techniques alone, which rely on generalization and suppression, are insufficient to provide complete protection against uniqueness-based metrics like Singling Out. The risk is only fully mitigated, dropping to 0.0, after the Synthetic Data Generation step. This confirms that the final layer of stochasticity and distribution learning provided by the synthesizer is essential to fully defend against this type of privacy attack.

To rigorously assess the value added by our multi-stage pipeline, we introduce a metric that quantifies the percentage improvement relative to using the Synthetic Data Generator (SDG) alone. This improvement metric is calculated as follows:

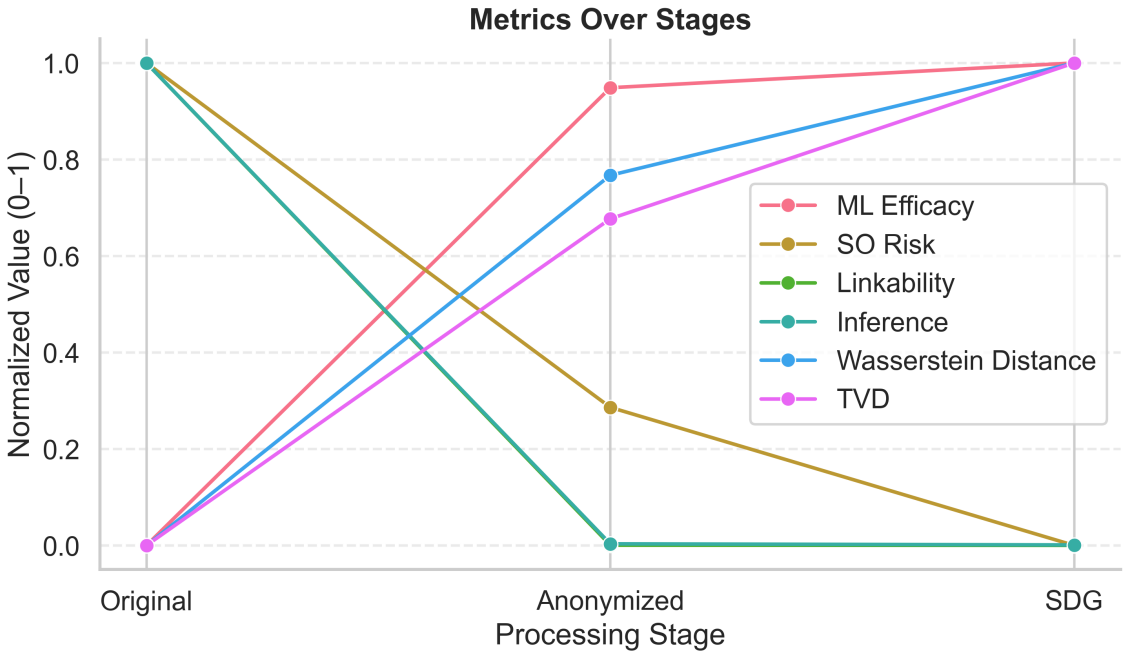


Figure 5.2: Evolution of the privacy, utility and fidelity metrics over the stages in our pipeline. *Original* indicates the score of the metrics when we use the training data as the synthetic dataset, *Anonymized* indicates the dataset after data anonymization and transition, and *SDG* denotes the results at the end of the pipeline. The results have been normalized using the Min-Max scaler. The results here are averaged on all the models and parameters.

$$R_{\text{pipeline}\to\text{SDG}} = \frac{R_{\text{pipeline}} - R_{\text{SDG}}}{R_{\text{SDG}}} \tag{5.3}$$

where R_{pipeline} represents the metric score achieved by our full pipeline, and R_{SDG} represents the score achieved by the SDG model trained directly on the original (un-anonymized) data baseline. The result is multiplied by 100 to express it as a percentage. A positive result indicates an improvement (for privacy metrics) or a degradation (for utility metrics) compared to the baseline.

Our initial analysis examines whether there are significant differences in pipeline performance based solely on the choice of the synthetic data generator. We report these results in Figure 5.3. As anticipated and consistent with the overall results from Figure 5.2, we generally observe an improvement on the privacy side (higher scores mean better privacy) and a corresponding degradation on the utility and fidelity side (higher scores mean worse utility/fidelity).

However, the plot reveals some interesting variations across the models:

- CTGAN results show the most robust improvement in protection against privacy attacks, but this comes at the cost of having the most substantial degradation in scores for both utility and fidelity metrics.
- REalTabFormer (RTF) and Tabddpm exhibit more complex behaviors. For RTF, the results align with expectations for utility and fidelity (worse scores in the pipeline),

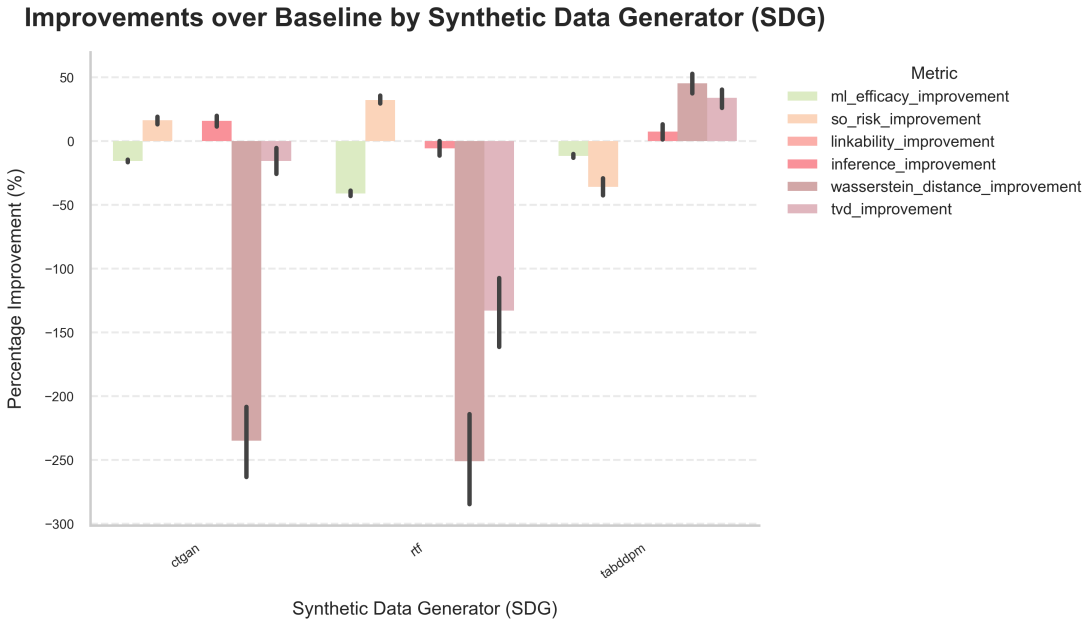


Figure 5.3: Comparison for privacy, utility and fidelity metrics grouping the results by synthesizer.

and it shows a clear improvement in the Singling Out attack. However, it displays a small degradation for the Inference metric, which is counterintuitive.

- Tabddpm presents the most unexpected results: the scores for fidelity metrics and Inference are actually better (lower percentage) than the SDG baseline, suggesting the pipeline structure somehow aided these specific outcomes. Conversely, the utility and Singling Out scores show the expected degradation.

To fully explain these unexpected and counterintuitive results—particularly the varying behavior of RTF and Tabddpm—we must delve deeper into the parameter space of the pipeline. The current plot represents only the average results for each synthesizer without isolating the crucial impact of other hyperparameters, such as the chosen k for data anonymization, the sampling type, or the transition type. Understanding these interactions is necessary to accurately diagnose the root cause of the observed metric behaviors. To improve the visual clarity of future plots, we will exclude the results of the fidelity metrics, although we refer the reader to the Appendix for supplemental experiments A.4.

Our current experimental focus includes an analysis of two transition order configurations: POST-transition utilizing only the uniform method, and PRE-transition where all three transition types—Single, Uniform, and Conditional—are tested before the Synthetic Data Generation (SDG) step.
Examining the results presented in Figure 5.4, we observe that the scores align perfectly with our theoretical expectations when considering the Single transition and the Uniform transition. In both cases, the inclusion of these transition steps in the pipeline leads to a clear improvement in the privacy metrics (higher scores compared to the SDG baseline), which confirms their effectiveness as an additional layer of protection. This

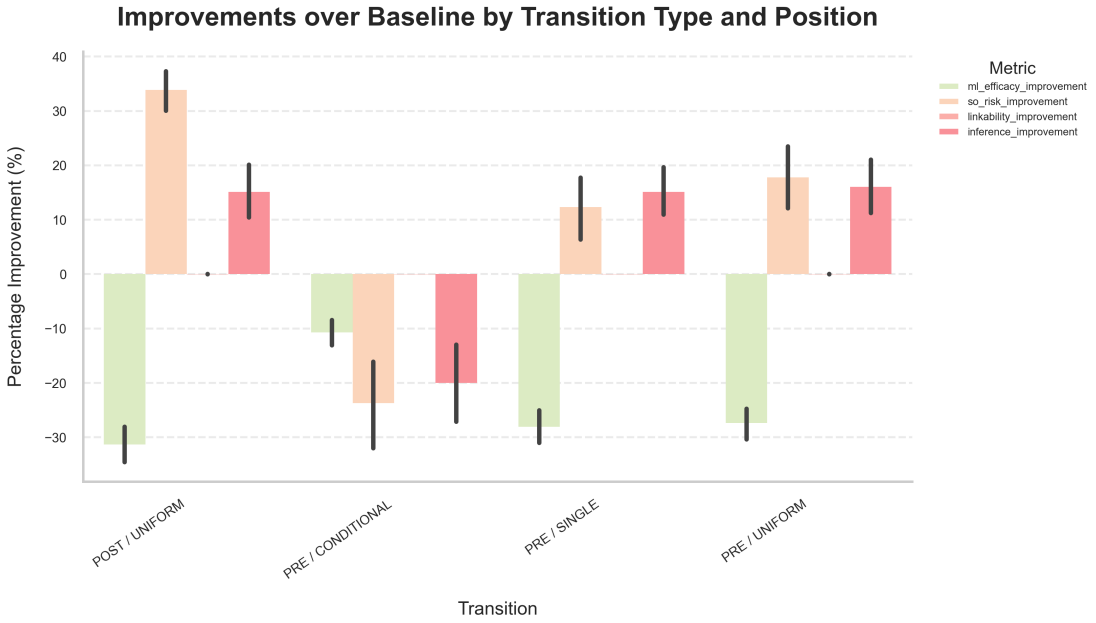


Figure 5.4: Comparison for privacy, utility and fidelity metrics grouping the results by transition.

privacy gain, however, is coupled with the anticipated trade-off: we report worse results in the utility metrics compared to the SDG-only baseline, as the generalization and transition processes inevitably introduce noise that degrades the data's utility.

However, the results for the Conditional transition tell a different story. For this high-fidelity method, the scores for all metrics are paradoxically better if we simply use SDG alone as the baseline. This means the inclusion of the Conditional transition in the pipeline actually degraded performance across the board. This outcome can be explained by a dual adverse effect. On the one hand, the utility of the data is severely damaged by the initial data anonymization phase, making the subsequent SDG process less effective. On the other hand, the very nature of the conditional transition—which samples attribute values from the internal, empirical conditional distribution of the equivalence class—appears to inject information that is too specific back into the dataset. By restoring these realistic local distributions, the data becomes susceptible to privacy attacks again, causing the privacy metrics to worsen and leading to a pipeline result that is inferior to the simple SDG baseline. This highlights the delicate balance between utility restoration and re-identification risk when using high-fidelity transition methods.

We investigated the impact of varying the k parameter within the k-anonymity privacy model on our final risk scores. Our observation is that the difference between the reported scores for the different k values is minimal.

If we were to calculate the confidence intervals for the results obtained under various k values, it is visually apparent that they almost perfectly overlap. This leads us to conclude that the value chosen for the k parameter has no significant impact on the final outcome of the pipeline as currently implemented. The effectiveness of the subsequent protective steps, specifically β-sampling, the transition step, and the robust distribu-

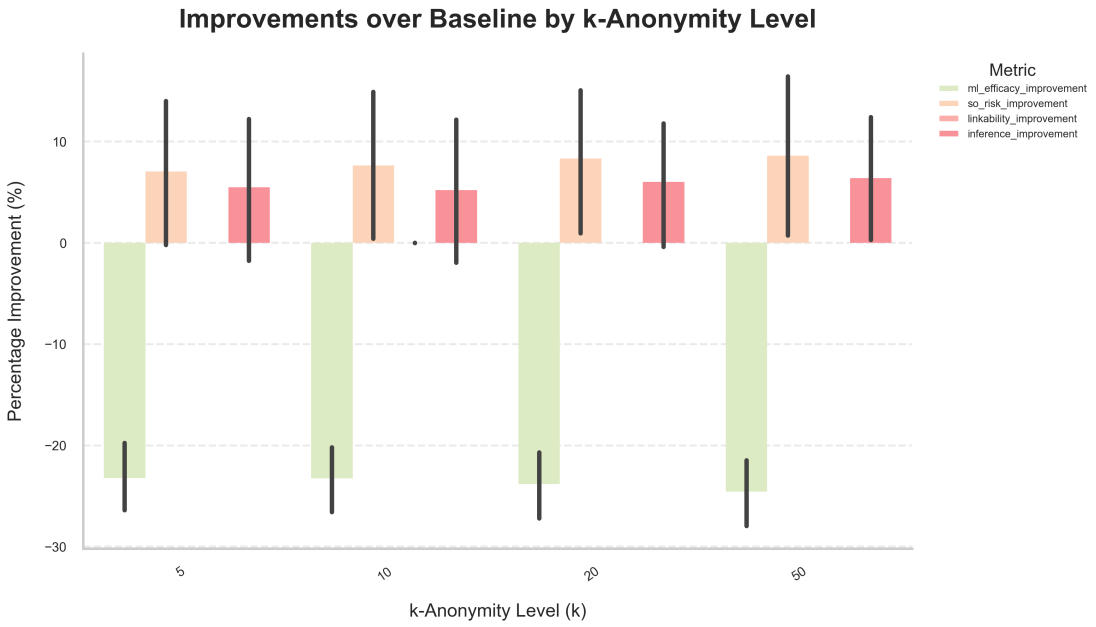


Figure 5.5: Comparison for privacy, utility and fidelity metrics grouping the results by anonymity level k.

tional learning of the Synthetic Data Generator—appears to dominate any marginal privacy gain achieved by increasing the initial k value, rendering the specific choice of k largely irrelevant in the full multi-layered framework.

In addition, we inspect the differences in the results with respect to the sampling method and parameters (see Figure 5.6). A consistent trend emerges where all tested defense mechanisms, including RANDOM and STRATIFIED, result in a negative utility improvement (i.e., a loss of ML efficacy, with values dropping as low as $\sim 30\%$) compared to the baseline, directly correlating with the strength of the defense: as the post-sampling rate increases from 0.1 to 1, the utility loss worsens while the privacy risk reduction (indicated by positive values for the risk metrics) simultaneously increases, often reaching gains of 10% to 20% in security. This inverse relationship highlights that while all methods are highly effective at mitigating privacy risks, the optimal choice for deployment will necessitate a careful balancing act, likely settling on an intermediate rate (e.g., 0.5) that provides acceptable privacy gains without incurring an excessive, detrimental drop in the synthetic data's utility for downstream machine learning tasks.

Finally, we examine the trade-off between privacy and utility, focusing specifically on Singling Out Risk and ML Utility. The results across multiple experimental runs are presented in Figure 5.7. Note that the metrics have been adjusted so that lower values indicate better performance for both axes. As the scatter plots demonstrate, our hybrid pipeline consistently improves privacy, yielding a lower singling-out risk while experiencing only a minor degradation in machine learning utility.

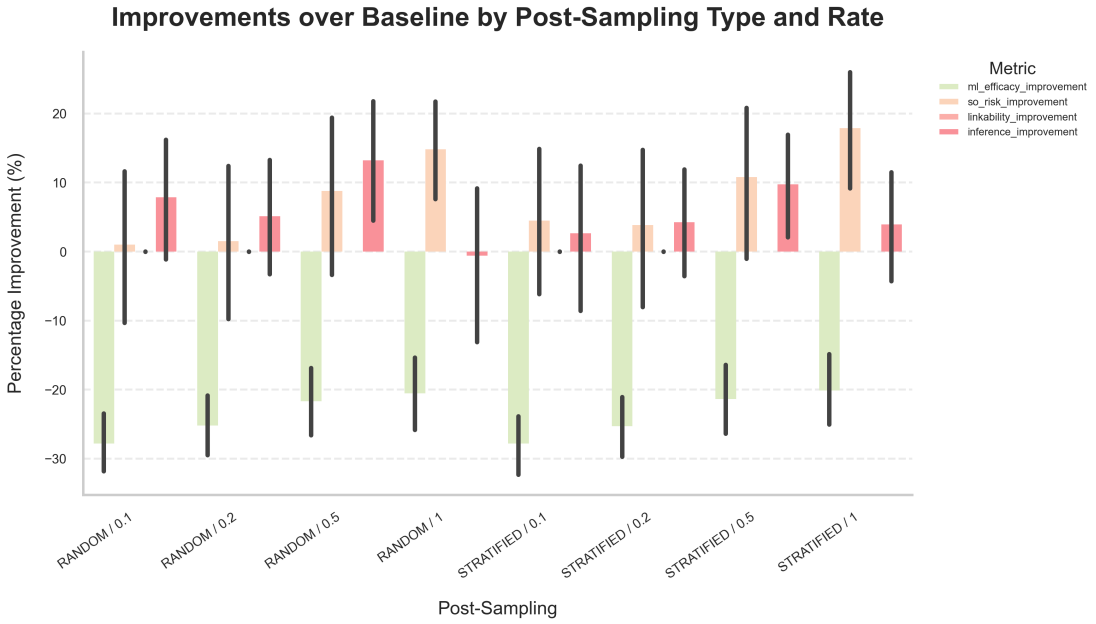


Figure 5.6: Comparison for privacy, utility and fidelity metrics grouping the results by sampling method.

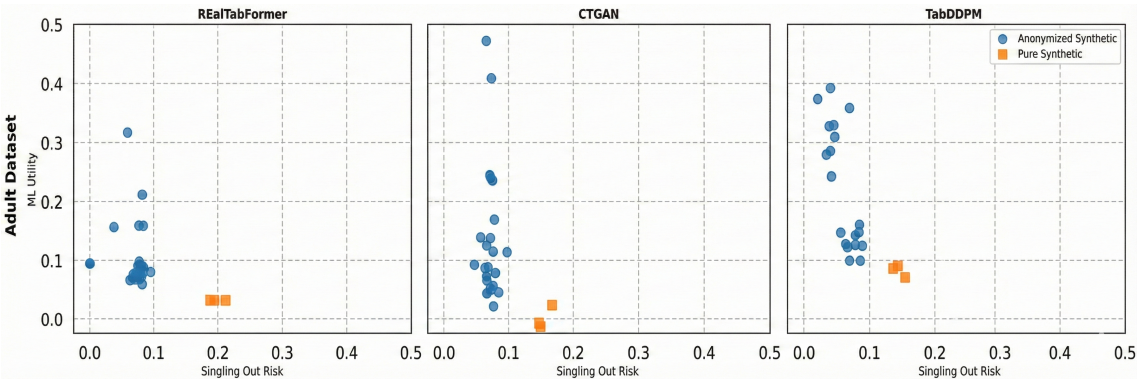


Figure 5.7: Comparison for Singling Out and ML Utility grouping the results by synthesizer.

5.4 Conclusions and future refinements

In this work, we have presented a novel, multi-stage approach that combines traditional data anonymization techniques with modern Synthetic Data Generation (SDG) models. As anticipated, the final output of our pipeline occupies a middle ground between a dataset resulting from pure data anonymization and one generated by SDG alone. On the one hand, we significantly increase the privacy guarantees by incorporating the initial anonymization phase, which deliberately degrades data utility through generalization and suppression. On the other hand, the combination of the transition step and the subsequent SDG phase is expected to restore a degree of utility to the final dataset. We anticipate that this improvement comes at the expense of slightly compromising the initial strict privacy protection, introducing a theoretical utility-privacy trade-off that warrants further investigation.
By systematically combining data anonymization, sampling, and synthetic data generation, our approach gains theoretical grounding. According to established works in formal privacy literature [23, 81], this sequencing, where anonymization and sampling are performed first, allows us to achieve a less strict definition of differential privacy (e.g., Group Differential Privacy). This theoretical claim holds if we consider both the transition step and the SDG process collectively as a single, complex "randomizer algorithm". This connection demands a deeper analytical inspection to quantify precisely how the different types of transition used may influence or violate the final privacy bounds. If this theoretical guarantee can be formally proven for our specific implementation, the entire pipeline would possess robust theoretical assurances that SDG alone, due to its inherent stochastic behavior, cannot provide.

Future Work

The immediate future work involves correctly displaying and analyzing the entirety of the experimental results. The plots presented in the previous section only show averaged scores, failing to isolate the specific impact of all interacting hyperparameters. A deeper, multivariate analysis is required to precisely understand the contribution of every single hyperparameter. It is highly plausible that a specific combination of a transition type and a value for the k parameter might yield an optimal trade-off between privacy and utility/fidelity. Furthermore, the selection of the synthetic data generator may have a disproportionately high impact on the final results when other parameters, such as the sampling type or the transition type, are fixed.
Another area for future exploration is the transition step itself. While we proposed and tested three distinct ways of performing the transition, such as Single, Uniform, and Conditional, many more variations exist. We observed that the Single and Uniform approaches excel at protecting privacy but sacrifice utility, whereas the Conditional transition unexpectedly degrades both. We believe that improving this step is paramount for optimizing the pipeline's performance. Future work should focus on exploring new transition methods that occupy a desirable middle ground between these two extremes. Moreover, a critical improvement could involve using more sophisticated AI models to accurately estimate the underlying probability function for sampling attribute values. While this may potentially degrade the final privacy results even more than the Conditional transition, it could serve as a valuable and robust extreme scenario to counteract

the initial simple randomness introduced by the Single and Uniform transitions.
A crucial last step for future validation involves a direct comparison of our pipeline's performance against a standard Differential Privacy-based Synthetic Data Generation (DP-SDG) approach. The fundamental difference between the two lies in the scope of noise application: DP-SDG adds calibrated noise to every single attribute in the dataset (or to the mechanism used to query the data), ensuring a rigorous privacy guarantee for the entire database. Conversely, our pipeline focuses the primary noise injection (through generalization, suppression, and sampling) specifically and only on the Quasi-Identifiers. This targeted noise application in our method is a deliberate effort to maximize data utility, as it leaves the non-QI attributes (the sensitive attributes themselves) relatively untouched, while still aiming for a form of group-level privacy and plausible deniability. Furthermore, we intend to evaluate the incremental benefit of our full pipeline by benchmarking its performance against a baseline consisting only of traditional anonymization techniques. This will allow us to quantify the specific impact of the subsequent synthesis stage on both data utility and privacy resilience. Analyzing the privacy-utility trade-offs between these fundamentally different noise strategies will be an important component of future work.

Chapter 6

Conclusions

This thesis undertook a comprehensive formal and empirical investigation into the fundamental conflict between data utility and privacy protection in the domain of tabular synthetic data generation (SDG). By systematically analyzing and advancing methodologies for privacy quantification and data synthesis, we have achieved three core objectives: enhancing the sensitivity of privacy auditing, standardizing the evaluation process, and engineering a novel, layered privacy pipeline.

On Attack-based Privacy Metrics (Chapter 3): Our proposal to incorporate a Contrastive Learning-based approach significantly enhanced the sensitivity of the Singling Out attack, particularly in detecting memorized outliers that resist traditional heuristic-based searches. The experimental results demonstrated that by operating in a semantically meaningful embedding space, our approach could often surpass the performance of the baseline Anonymeter framework and successfully track increasing risk in the Noisy Synthesizer and Overfitting Risk Models. Furthermore, we rigorously evaluated the limitations of applying this computationally complex method to Linkability and Attribute Inference attacks, noting that the added statistical gain did not, in all cases, justify the substantial computational cost incurred. This highlights the necessity of balancing methodological rigor with practical efficiency in real-world privacy auditing.

On the Taxonomy and Metric Standardization (Chapter 4): The proposed three-pillar Taxonomy proved highly effective for organizing the disparate landscape of synthetic data privacy quantification into coherent categories: Privacy Properties, Statistical Indicators, and Attack Simulations. Our empirical analysis using the newly defined Risk Models revealed a strong, near-perfect correlation between multiple No-box metrics (e.g., IMS, DCR, Singling Out, GTCAP) under the controlled Leaky and Overfitting scenarios. This significant finding implies a profound convergence in what these metrics are actually measuring: the localized memorization and statistical proximity between the real and synthetic data at the record level. This suggests that choosing a metric for routine auditing might be better guided by the criteria of robustness and efficiency (favoring quicker Statistical Indicators like DCR) rather than by the complexity of the underlying attack simulation. Conversely, the metrics in the DP risk model showed negligible responsiveness to changes in the privacy budget, a phenomenon attributed to the overriding influence of the low utility induced by DP noise.

On the Hybrid Synthesis Pipeline (Chapter 5): The Hybrid Data Synthesis Pipeline successfully demonstrated its core conceptual value: enforcing formal privacy a priori while leveraging SDG to restore utility. The results showed that the initial Anonymization phase accounted for the vast majority of privacy risk reduction (e.g., Inference risk immediately dropped to 0.0). The final SDG step was indispensable for fully mitigating remaining risks, such as Singling Out, and further smoothing out utility degradation. A critical trade-off was confirmed in the Transition Step: methods like Single and Uniform successfully bolstered privacy at the expense of utility, while the high-fidelity Conditional transition paradoxically worsened performance, suggesting it re-injected overly specific information and exceeded the acceptable trade-off boundary. The minimal impact of increasing the k-anonymity parameter confirmed that the layered protection offered by subsequent steps (β-sampling, Transition, SDG) is the dominant factor in final risk reduction.

6.1 Future Directions

The research conducted in this thesis paves the way for several critical avenues of future exploration:

- Algorithmic Efficiency of Contrastive Attacks: The primary limitation of our advanced Linkability and Attribute Inference attacks remains their prohibitive computational cost. Future work must focus on optimizing the neural network training and nearest-neighbor search within the latent space to make these high-sensitivity auditing tools practical for large-scale datasets.

- Refinement of the Utility-Preserving Transition: The Transition Step in the hybrid pipeline is the single most critical area for refinement. Future research must focus on exploring novel transition mechanisms that strategically sample or estimate attribute values to occupy the optimal utility-privacy trade-off space, avoiding the over-specificity observed with the Conditional transition. This may involve exploring sophisticated AI models for localized probability estimation without memorization.

- Formal Privacy Guarantees for Hybridization: A crucial theoretical step is to formally prove how the sequence of Anonymization, Sampling, Transition, and SDG translates into a verifiable privacy bound, such as a form of Group Differential Privacy. This would provide a robust, mathematical safety certification that non-DP synthetic data currently lacks.

- Targeted Noise Strategies: A final comparative study between our pipeline's strategy (noise targeted only at Quasi-Identifiers) and traditional DP-SDG (noise applied universally) is necessary. Analyzing the privacy-utility trade-offs of these fundamentally different noise-application scopes will define the next generation of best practices for privacy-preserving data synthesis.

List of Figures

List of Tables

Bibliography

[1] Steven ruggles, sarah flood, ronald goeken, josiah grover, erin meyer, jose pacas, and matthew sobek. ipums usa: Version 8.0 extract of 1940 census for u.s. census bureau disclosure avoidance research. minneapolis, mn: Ipums, 2008. DOI: https://doi.org/10.18128/D010.V8.0.EXT1940USCB.

[2] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318, 2016.

[3] C. C. Aggarwal. On k-anonymity and the curse of dimensionality. In *VLDB*, volume 5, pages 901–909, 2005.

[4] M. S. M. S. Annamalai, A. Gadotti, and L. Rocher. A linear reconstruction approach for attribute inference attacks against synthetic data, 2024. URL https://arxiv.org/abs/2301.10053.

[5] Article 29 Data Protection Working Party. Available at: https://ec.europa.eu/justice/article-29/documentation/opinion-recommendation/files/2014/wp216_en.pdf, year = 2014.

[6] B. Balle, G. Barthe, and M. Gaboardi. Privacy amplification by subsampling: Tight analyses via couplings and divergences. *Advances in neural information processing systems*, 31, 2018.

[7] B. Becker and R. Kohavi. Adult. UCI Machine Learning Repository, 1996. DOI: https://doi.org/10.24432/C5XW20.

[8] A. Beduschi. Synthetic data protection: Towards a paradigm change in data regulation? *Big Data & Society*, 11(1):20539517241231277, 2024. doi: 10.1177/20539517241231277. URL https://doi.org/10.1177/20539517241231277.

[9] S. M. Bellovin, K. Dutta, Preetam, and N. Reitinger. Privacy and synthetic datasets. *Stanford Technology Law Review*, 2018.

[10] Y. Bengio et al. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2(1):1–127, 2009.

[11] V. Bindschaedler, R. Shokri, and C. A. Gunter. Plausible deniability for privacy-preserving data synthesis. *arXiv preprint arXiv:1708.07975*, 2017.

[12] V. Borisov, K. Seßler, T. Leemann, M. Pawelczyk, and G. Kasneci. Language models are realistic tabular data generators. *arXiv preprint arXiv:2210.06280*, 2022.

[13] A. Boudewijn, A. F. Ferraris, D. Panfilo, V. Cocca, S. Zinutti, K. De Schepper, and C. R. Chauvenet. Privacy measurement in tabular synthetic data: State of the art and future research directions. *arXiv preprint arXiv:2311.17453*, 2023.

[14] N. Brandes, D. Ofer, Y. Peleg, N. Rappoport, and M. Linial. Proteinbert: a universal deep-learning model of protein sequence and function. *Bioinformatics*, 38(8):2102–2110, 2022.

[15] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander. Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pages 93–104, 2000.

[16] J. Brickell and V. Shmatikov. The cost of privacy: destruction of data-mining utility in anonymized data publishing. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 70–78, 2008.

[17] R. Burke, A. Felfernig, and M. H. Göker. Recommender systems: An overview. *Ai Magazine*, 32(3):13–18, 2011.

[18] J. A. Calandrino, A. Kilzer, A. Narayanan, E. W. Felten, and V. Shmatikov. " you might also like:" privacy risks of collaborative filtering. In *2011 IEEE symposium on security and privacy*, pages 231–246. IEEE, 2011.

[19] N. Carlini, C. Liu, Ú. Erlingsson, J. Kos, and D. Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX security symposium (USENIX security 19)*, pages 267–284, 2019.

[20] N. Carlini, M. Jagielski, C. Zhang, N. Papernot, A. Terzis, and F. Tramer. The privacy onion effect: Memorization is relative, 2022. URL https://arxiv.org/abs/2206.10469.

[21] Center for Health Statistics Texas Department of State Health Services. Texas hospital discharge data public use data., 2005. https://www.dshs.texas.gov/texas-health-care-information-collection/general-public-information/hospital-discharge-data-public.

[22] R. Chapelle and B. Falissard. Statistical properties and privacy guarantees of an original distance-based fully synthetic data generation method, 2023. URL https://arxiv.org/abs/2310.06571.

[23] K. Chaudhuri and N. Mishra. When random sampling preserves privacy. In *Annual International Cryptology Conference*, pages 198–213. Springer, 2006.

[24] D. Chen, N. Yu, Y. Zhang, and M. Fritz. Gan-leaks: A taxonomy of membership inference attacks against generative models. In *Proceedings of the 2020 ACM SIGSAC conference on computer and communications security*, pages 343–362, 2020.

[25] T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[26] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In H. D. III and A. Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/chen20j.html.

[27] Y. Chen, J. Taub, and M. Elliot. The trade-off between information utility and disclosure risk in a ga synthetic data generator. *Joint UNECE/Eurostat Work Session on Statistical Data Confidentiality*, 2019.

[28] E. Choi, S. Biswal, B. Malin, J. Duke, W. F. Stewart, and J. Sun. Generating multi-label discrete patient records using generative adversarial networks. In *Machine learning for healthcare conference*, pages 286–305. PMLR, 2017.

[29] N. De Cao, W. Aziz, and I. Titov. Block neural autoregressive flow. In *Uncertainty in artificial intelligence*, pages 1263–1273. PMLR, 2020.

[30] J. Domingo-Ferrer and V. Torra. A quantitative comparison of disclosure control methods for microdata. In *Confidentiality, disclosure and data access: theory and practical applications for statistical agencies*, number North-Holland, pages 111–134, 2001.

[31] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[32] C. Dwork. Differential privacy. In *International colloquium on automata, languages, and programming*, pages 1–12. Springer, 2006.

[33] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor. Our data, ourselves: Privacy via distributed noise generation. In *Annual international conference on the theory and applications of cryptographic techniques*, pages 486–503. Springer, 2006.

[34] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006.

[35] C. Dwork, A. Roth, et al. The algorithmic foundations of differential privacy. *Foundations and trends® in theoretical computer science*, 9(3–4):211–407, 2014.

[36] G. D'Acquisto, A. Cohen, M. Naldi, and K. Nissim. From isolation to identification. In *International Conference on Privacy in Statistical Databases*, pages 3–17. Springer, 2024.

[37] C. Esteban, S. L. Hyland, and G. Rätsch. Real-valued (medical) time series generation with recurrent conditional gans. *arXiv preprint arXiv:1706.02633*, 2017.

[38] European Parliament and Council of the European Union. Regulation (EU) 2016/679 of the European Parliament and of the Council, . URL https://data.europa.eu/eli/reg/2016/679/oj.

[39] European Parliament and Council of the European Union. Regulation (EU) 2016/679 of the European Parliament and of the Council - Recital 26, . URL https://www.privacy-regulation.eu/en/recital-26-GDPR.htm.

[40] M. L. Fang, D. S. Dhami, and K. Kersting. Dp-ctgan: Differentially private medical data generation using ctgans. In *International conference on artificial intelligence in medicine*, pages 178–188. Springer, 2022.

[41] M. Fredrikson, E. Lantz, S. Jha, S. Lin, D. Page, and T. Ristenpart. Privacy in pharmacogenetics: An {End-to-End} case study of personalized warfarin dosing. In *23rd USENIX security symposium (USENIX Security 14)*, pages 17–32, 2014.

[42] B. C. Fung, K. Wang, R. Chen, and P. S. Yu. Privacy-preserving data publishing: A survey of recent developments. *ACM Computing Surveys (Csur)*, 42(4):1–53, 2010.

[43] M. S. Gal and O. Lynskey. Synthetic data: Legal implications of the data-generation revolution. *Iowa Law Review*, 109(1087):1087–1154, 2024.

[44] G. Ganev and E. D. Cristofaro. On the inadequacy of similarity-based privacy metrics: Reconstruction attacks against "truly anonymous synthetic data", 2023.

[45] S. R. Ganta, S. P. Kasiviswanathan, and A. Smith. Composition attacks and auxiliary information in data privacy. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 265–273, 2008.

[46] R. C. Geyer, T. Klein, and M. Nabi. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*, 2017.

[47] Z. Ghahramani. Probabilistic machine learning and artificial intelligence. *Nature*, 521(7553):452–459, 2015.

[48] B. Ghojogh and A. Ghodsi. Diffusion Models: Tutorial and Survey. working paper or preprint, July 2024. URL https://hal.science/hal-04642649.

[49] M. Giomi, F. Boenisch, C. Wehmeyer, and B. Tasnádi. A unified framework for quantifying privacy risk in synthetic data. *arXiv preprint arXiv:2211.10459*, 2022.

[50] S. Golob. *Privacy Vulnerabilities in Marginals-based Synthetic Data*. PhD thesis, 2024.

[51] A. Goncalves, P. Ray, B. Soper, J. Stevens, L. Coyle, and A. P. Sales. Generation and evaluation of synthetic patient data. *BMC medical research methodology*, 20 (1):108, 2020.

[52] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[53] S. Gopi, Y. T. Lee, and L. Wutschitz. Numerical composition of differential privacy. *Advances in Neural Information Processing Systems*, 34:11631–11642, 2021.

[54] J. C. Gower. A general coefficient of similarity and some of its properties. *Biometrics*, pages 857–871, 1971.

[55] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola. A kernel two-sample test. *The journal of machine learning research*, 13(1):723–773, 2012.

[56] J. Hayes, L. Melis, G. Danezis, and E. D. Cristofaro. Logan: Membership inference attacks against generative models, 2018.

[57] D. Heckerman. A tutorial on learning with bayesian networks. *Learning in graphical models*, pages 301–354, 1998.

[58] B. Hilprecht, M. Härterich, and D. Bernau. Monte carlo and reconstruction membership inference attacks against generative models. *Proceedings on Privacy Enhancing Technologies*, 2019.

[59] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[60] E. Hoogeboom, D. Nielsen, P. Jaini, P. Forré, and M. Welling. Argmax flows and multinomial diffusion: Learning categorical distributions. *Advances in neural information processing systems*, 34:12454–12465, 2021.

[61] K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[62] F. Houssiau, J. Jordon, S. N. Cohen, O. Daniel, A. Elliott, J. Geddes, C. Mole, C. Rangel-Smith, and L. Szpruch. Tapas: a toolbox for adversarial privacy auditing of synthetic data. *arXiv preprint arXiv:2211.06550*, 2022.

[63] J. Hradec, M. Craglia, M. Di Leo, S. De Nigris, N. Ostlaender, and N. Nicholson. *Multipurpose synthetic population for policy applications*. JRC technical report, 2022.

[64] H. Hu and J. Pang. Membership inference attacks against gans by leveraging overrepresentation regions. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, CCS '21, page 2387–2389, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450384544. doi: 10.1145/3460120.3485338. URL https://doi.org/10.1145/3460120.3485338.

[65] H. Hu, Z. Salcic, L. Sun, G. Dobbie, and X. Zhang. Source inference attacks in federated learning. In *2021 IEEE International Conference on Data Mining (ICDM)*, pages 1102–1107. IEEE, 2021.

[66] J. Hu and C. M. Bowen. Advancing microdata privacy protection: A review of synthetic data, 2023.

[67] V. Hudovernik. Relational data generation with graph neural networks and latent diffusion models. In *NeurIPS 2024 Third Table Representation Learning Workshop*, 2024.

[68] A. J. Jeckmans, M. Beye, Z. Erkin, P. Hartel, R. L. Lagendijk, and Q. Tang. Privacy in recommender systems. In *Social media retrieval*, pages 263–281. Springer, 2012.

[69] J. Jordon, J. Yoon, and M. Van Der Schaar. Pate-gan: Generating synthetic data with differential privacy guarantees. In *International conference on learning representations*, 2018.

[70] B. Kaabachi, J. Despraz, T. Meurers, K. Otte, M. Halilovic, B. Kulynych, F. Prasser, and J. L. Raisaro. A scoping review of privacy and utility metrics in medical synthetic data. *NPJ digital medicine*, 8(1):60, 2025.

[71] J. Kim, C. Lee, and N. Park. Stasy: Score-based tabular data synthesis. *arXiv preprint arXiv:2210.04018*, 2022.

[72] A. Kotelnikov, D. Baranchuk, I. Rubachev, and A. Babenko. Tabddpm: Modelling tabular data with diffusion models. In *International conference on machine learning*, pages 17564–17579. PMLR, 2023.

[73] S. Kullback and R. A. Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[74] R. S. S. Kumar, D. O. Brien, K. Albert, S. Viljöen, and J. Snover. Failure modes in machine learning systems, 2019.

[75] P. Le and W. Zuidema. Quantifying the vanishing gradient and long distance dependency problem in recursive neural networks and recursive lstms. *arXiv preprint arXiv:1603.00423*, 2016.

[76] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[77] M. Lee. Gelu activation function in deep learning: a comprehensive mathematical analysis and performance. *arXiv preprint arXiv:2305.12073*, 2023.

[78] J. Li, J.-J. Yang, Y. Zhao, B. Liu, M. Zhou, J. Bi, and Q. Wang. Enforcing differential privacy for shared collaborative filtering. *IEEE Access*, 5:35–49, 2016.

[79] L. Li, Y. Fan, M. Tse, and K.-Y. Lin. A review of applications in federated learning. *Computers & Industrial Engineering*, 149:106854, 2020.

[80] N. Li, T. Li, and S. Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *2007 IEEE 23rd international conference on data engineering*, pages 106–115. IEEE, 2006.

[81] N. Li, W. Qardaji, and D. Su. On sampling, anonymization, and differential privacy or, k-anonymization meets differential privacy. In *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security*, pages 32–33, 2012.

[82] J. Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151, 2002.

[83] C. A. López and A. Elbi. On the legal nature of synthetic data. *Proceedings of the Thirty-sixth Conference on Neural Information Processing Systems (NeurIPS)*, 2022.

[84] C. A. F. López et al. On the legal nature of synthetic data. In *NeurIPS 2022 Workshop on Synthetic Data for Empowering ML Research*, 2022.

[85] D. Lopez-Paz and M. Oquab. Revisiting classifier two-sample tests. *arXiv preprint arXiv:1610.06545*, 2016.

[86] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. *Acm transactions on knowledge discovery from data (tkdd)*, 1(1):3–es, 2007.

[87] F. J. Massey Jr. The kolmogorov-smirnov test for goodness of fit. *Journal of the American statistical Association*, 46(253):68–78, 1951.

[88] W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.

[89] R. McKenna, B. Mullins, D. Sheldon, and G. Miklau. Aim: An adaptive and iterative mechanism for differentially private synthetic data. *arXiv preprint arXiv:2201.12677*, 2022.

[90] H. B. McMahan, D. Ramage, K. Talwar, and L. Zhang. Learning differentially private recurrent language models. *arXiv preprint arXiv:1710.06963*, 2017.

[91] M. Meeus, F. Guepin, A.-M. Cretu, and Y.-A. de Montjoye. Achilles' heels: Vulnerable record identification in synthetic data publishing, 2023.

[92] A. Meyerson and R. Williams. On the complexity of optimal k-anonymity. In *Proceedings of the twenty-third ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 223–228, 2004.

[93] I. Mironov. Rényi differential privacy. In *2017 IEEE 30th computer security foundations symposium (CSF)*, pages 263–275. IEEE, 2017.

[94] A. Narayanan and V. Shmatikov. Robust de-anonymization of large sparse datasets. In *2008 IEEE Symposium on Security and Privacy (sp 2008)*, pages 111–125. IEEE, 2008.

[95] B. Nowok, G. M. Raab, and C. Dibben. synthpop: Bespoke creation of synthetic data in r. *Journal of statistical software*, 74:1–26, 2016.

[96] K. O'shea and R. Nash. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*, 2015.

[97] I. Padhi, Y. Schiff, I. Melnyk, M. Rigotti, Y. Mroueh, P. Dognin, J. Ross, R. Nair, and E. Altman. Tabular transformers for modeling multivariate time series. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3565–3569. IEEE, 2021.

[98] M. N. P. Palacios, A. Boudewijn, S. Saccani, A. F. Ferraris, D. Sofronieva, G. D'Acquisto, F. Brozzetti, D. Panfilo, and L. Bortolussi. Empirical evaluation of structured synthetic data privacy metrics: Novel experimental framework. *arXiv preprint arXiv:2512.16284*, 2025.

[99] M. N. P. Palacios, S. Saccani, G. Sgroi, A. Boudewijn, and L. Bortolussi. Contrastive learning-based privacy metrics in tabular synthetic datasets. *arXiv preprint arXiv:2502.13833*, 2025.

[100] D. Panfilo, A. Boudewijn, S. Saccani, A. Coser, B. Svara, C. R. Chauvenet, C. A. Mami, and E. Medvet. A deep learning-based pipeline for the generation of synthetic tabular data. *IEEE Access*, pages 1–1, 2023. doi: 10.1109/ACCESS.2023.3288336.

[101] D. Panfilo et al. Generating privacy-compliant, utility-preserving synthetic tabular and relational datasets through deep learning. 2022.

[102] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[103] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[104] L. Pilgram, F. K. Dankar, J. Drechsler, M. Elliot, J. Domingo-Ferrer, P. Francis, M. Kantarcioglu, L. Kong, B. Malin, K. Muralidhar, et al. A consensus privacy metrics framework for synthetic data. *Patterns*, 6(10), 2025.

[105] M. Platzer and T. Reutterer. Holdout-based empirical assessment of mixed-type synthetic data. *Frontiers in big Data*, 4:679939, 2021.

[106] F. Prasser, F. Kohlmayer, R. Lautenschläger, and K. A. Kuhn. Arx-a comprehensive tool for anonymizing biomedical data. In *AMIA Annual Symposium Proceedings*, volume 2014, page 984, 2014.

[107] Z. Qian, B.-C. Cebere, and M. van der Schaar. Synthcity: facilitating innovative use cases of synthetic data in different data modalities, 2023. URL https://arxiv.org/abs/2301.07573.

[108] G. M. Raab. Utility and disclosure risk for differentially private synthetic categorical data. In J. Domingo-Ferrer and M. Laurent, editors, *Privacy in Statistical Databases*, pages 250–265, Cham, 2022. Springer International Publishing. ISBN 978-3-031-13945-1.

[109] G. M. Raab, B. Nowok, and C. Dibben. Practical privacy metrics for synthetic data, 2024. URL https://arxiv.org/abs/2406.16826.

[110] T. E. Raghunathan, J. M. Lepkowski, J. Van Hoewyk, P. Solenberger, et al. A multivariate technique for multiply imputing missing values using a sequence of regression models. *Survey methodology*, 27(1):85–96, 2001.

[111] J. P. Reiter. Synthetic data: A look back and a look forward. *Trans. Data Priv.*, 16 (1):15–24, 2023.

[112] J. Ren, X. Xu, and H. Yu. Improved collaborative filtering algorithm incorporating user information and using differential privacy. In *CCF Conference on Computer Supported Cooperative Work and Social Computing*, pages 458–471. Springer, 2019.

[113] F. Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

[114] L. Rosenblatt, X. Liu, S. Pouyanfar, E. de Leon, A. Desai, and J. Allen. Differentially private synthetic data: Applied evaluations and enhancements, 2020.

[115] Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. *International journal of computer vision*, 40(2):99–121, 2000.

[116] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

[117] P. Samarati. Protecting respondents identities in microdata release. *IEEE transactions on Knowledge and Data Engineering*, 13(6):1010–1027, 2002.

[118] D. Scassola, S. Saccani, and L. Bortolussi. Graph conditional flow matching for relational data generation. *arXiv preprint arXiv:2505.15668*, 2025.

[119] R. M. Schmidt. Recurrent neural networks (rnns): A gentle introduction and overview. *arXiv preprint arXiv:1912.05911*, 2019.

[120] T. Shenkar and L. Wolf. Anomaly detection for tabular data with internal contrastive learning. In *International conference on learning representations*, 2022.

[121] R. Shokri, M. Stronati, C. Song, and V. Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pages 3–18. IEEE, 2017.

[122] A. V. Solatorio and O. Dupriez. Realtabformer: Generating realistic relational and tabular data using transformers. *arXiv preprint arXiv:2302.02041*, 2023.

[123] T. Stadler, B. Oprisanu, and C. Troncoso. Synthetic data-a privacy mirage. *arXiv preprint arXiv:2011.07018*, 2020.

[124] T. Stadler, B. Oprisanu, and C. Troncoso. Synthetic data – anonymisation groundhog day, 2022.

[125] M. Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the royal statistical society: Series B (Methodological)*, 36(2):111–133, 1974.

[126] L. Sweeney. k-anonymity: A model for protecting privacy. *International journal of uncertainty, fuzziness and knowledge-based systems*, 10(05):557–570, 2002.

[127] C. Task, K. Bhagat, and G. Howarth. SDNist v2: Deidentified Data Report Tool, Mar. 2023. URL https://data.nist.gov/od/id/mds2-2943.

[128] J. Taub, M. J. Elliot, and G. M. Raab. Creating the best risk-utility profile : The synthetic data challenge. 2019. URL https://api.semanticscholar.org/CorpusID:204747180.

[129] B. Van Breugel and M. Van der Schaar. Beyond privacy: Navigating the opportunities and challenges of synthetic data. *arXiv preprint arXiv:2304.03722*, 2023.

[130] B. van Breugel, H. Sun, Z. Qian, and M. van der Schaar. Membership inference attacks against synthetic data through overfitting detection, 2023.

[131] S. Van Buuren, J. P. Brand, C. G. Groothuis-Oudshoorn, and D. B. Rubin. Fully conditional specification in multivariate imputation. *Journal of statistical computation and simulation*, 76(12):1049–1064, 2006.

[132] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[133] D. Wang, N. Ding, P. Li, and H.-T. Zheng. Cline: Contrastive learning with semantic negative examples for natural language understanding. *arXiv preprint arXiv:2107.00440*, 2021.

[134] T. Wang and P. Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International conference on machine learning*, pages 9929–9939. PMLR, 2020.

[135] Y.-X. Wang, B. Balle, and S. P. Kasiviswanathan. Subsampled rényi differential privacy and analytical moments accountant. In *The 22nd international conference on artificial intelligence and statistics*, pages 1226–1235. PMLR, 2019.

[136] E. B. Wilson. Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22(158):209–212, 1927.

[137] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni. Modeling tabular data using conditional gan. *Advances in neural information processing systems*, 32, 2019.

[138] Z. Yao, N. Krčo, G. Ganev, and Y.-A. de Montjoye. The dcr delusion: Measuring the privacy risk of synthetic data. In *European Symposium on Research in Computer Security*, pages 469–487. Springer, 2025.

[139] S. Yeom, I. Giacomelli, A. Menaged, M. Fredrikson, and S. Jha. Overfitting, robustness, and malicious algorithms: A study of potential causes of privacy risk in machine learning. *Journal of Computer Security*, 28(1):35–70, 2020.

[140] A. Yousefpour, I. Shilov, A. Sablayrolles, D. Testuggine, K. Prasad, M. Malek, J. Nguyen, S. Ghosh, A. Bharadwaj, J. Zhao, et al. Opacus: User-friendly differential privacy library in pytorch. *arXiv preprint arXiv:2109.12298*, 2021.

[141] S. Zargar. Introduction to sequence learning models: Rnn, lstm, gru. *Department of Mechanical and Aerospace Engineering, North Carolina State University*, 37988518, 2021.

[142] J. Zhang, G. Cormode, C. M. Procopiuc, D. Srivastava, and X. Xiao. Privbayes: Private data release via bayesian networks. *ACM Transactions on Database Systems (TODS)*, 42(4):1–41, 2017.

[143] Y. Zhang, R. Jia, H. Pei, W. Wang, B. Li, and D. Song. The secret revealer: Generative model-inversion attacks against deep neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 253–261, 2020.

[144] J. Zhou, Y. Chen, C. Shen, and Y. Zhang. Property inference attacks against gans. *arXiv preprint arXiv:2111.07608*, 2021.

[145] T. Zhu, G. Li, Y. Ren, W. Zhou, and P. Xiong. Differential privacy for neighborhood-based collaborative filtering. In *Proceedings of the 2013 IEEE/ACM international conference on advances in social networks analysis and mining*, pages 752–759, 2013.

[146] A. Ziller, D. Usynin, R. Braren, M. Makowski, D. Rueckert, and G. Kaissis. Medical imaging deep learning with differential privacy. *Scientific Reports*, 11(1):13524, 2021.

Appendix A

Supplementary Material for Chapter 2

A.1 Complete experimental results

This section provides a comprehensive granular view of the performance for all privacy metrics evaluated across the three primary datasets: Adult, Texas, and Census. The objective was to observe how different attack-based and similarity-based methods respond to controlled levels of risk insertion 4.5. Most metrics demonstrate a strong linear response to direct data leakage in the leaky risk model. In the overfitting risk model, a monotonic increase in risk is observed as the generator trains beyond optimal generalization. Conversely, Differential Privacy (DP) generators consistently reported negligible risk sensitivity, likely because the excessive noise required for a tight privacy budget destroyed the data's utility, rendering them unvulnerable but also non-representative.

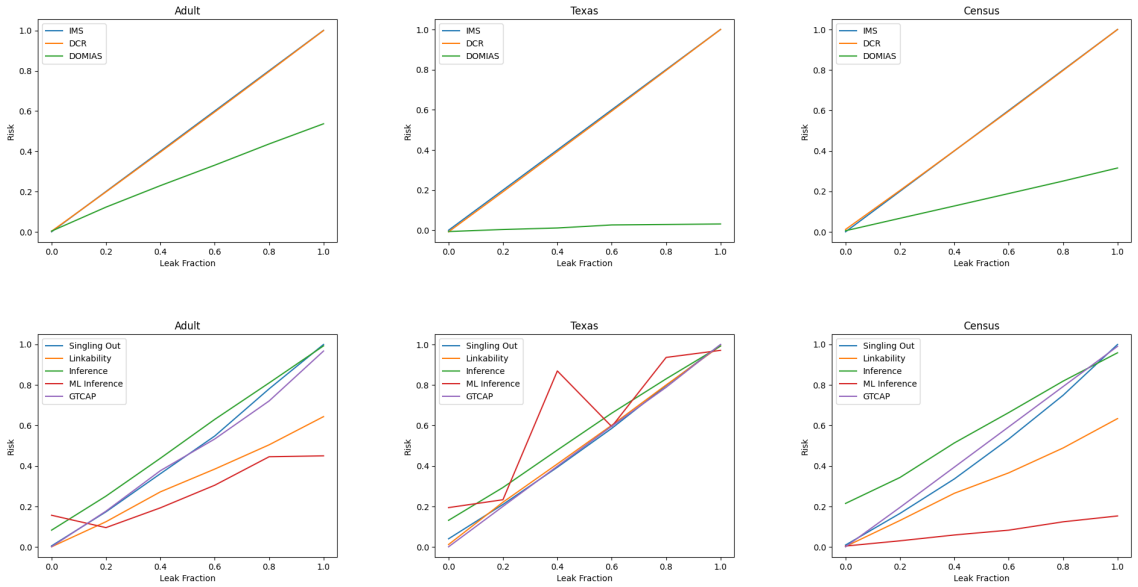


Figure A.1: Risk assessment methods evaluated using the leaky risk model

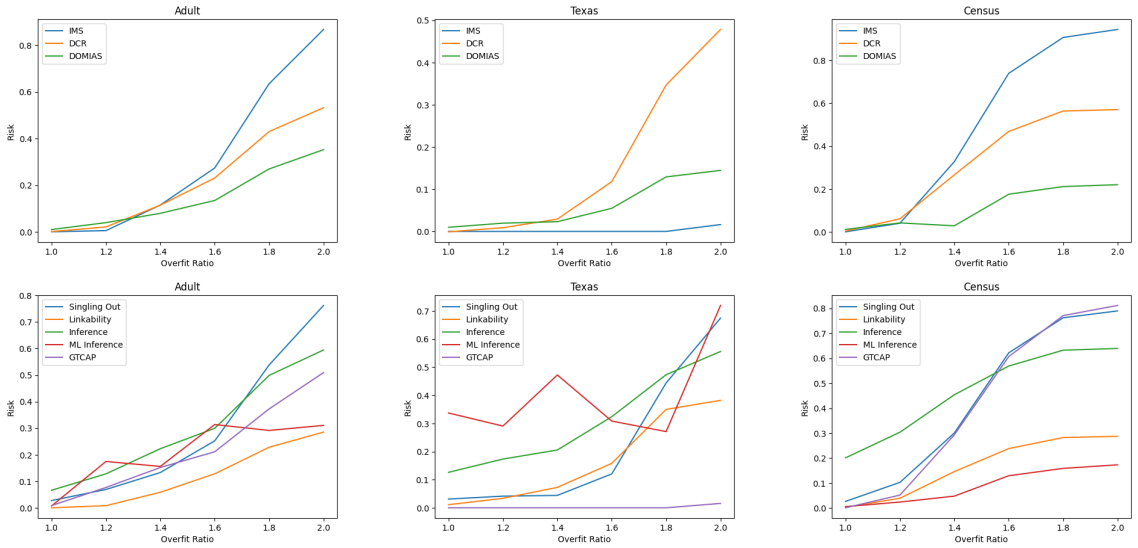


Figure A.2: Risk assessment methods evaluated using the overfit risk model - RTF [122]

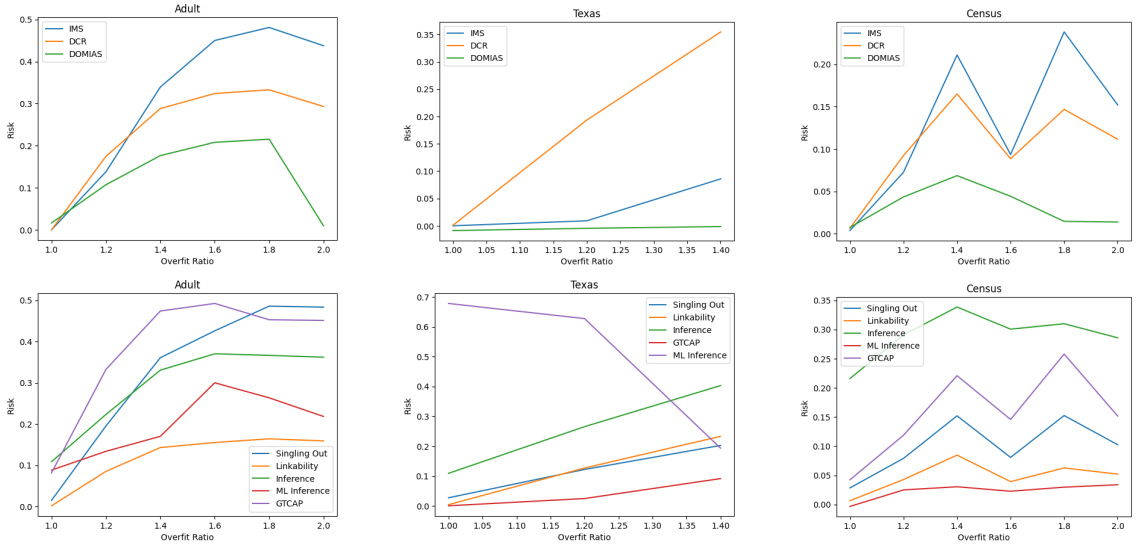


Figure A.3: Risk assessment methods evaluated using the overfit risk model - Synthpop [95]

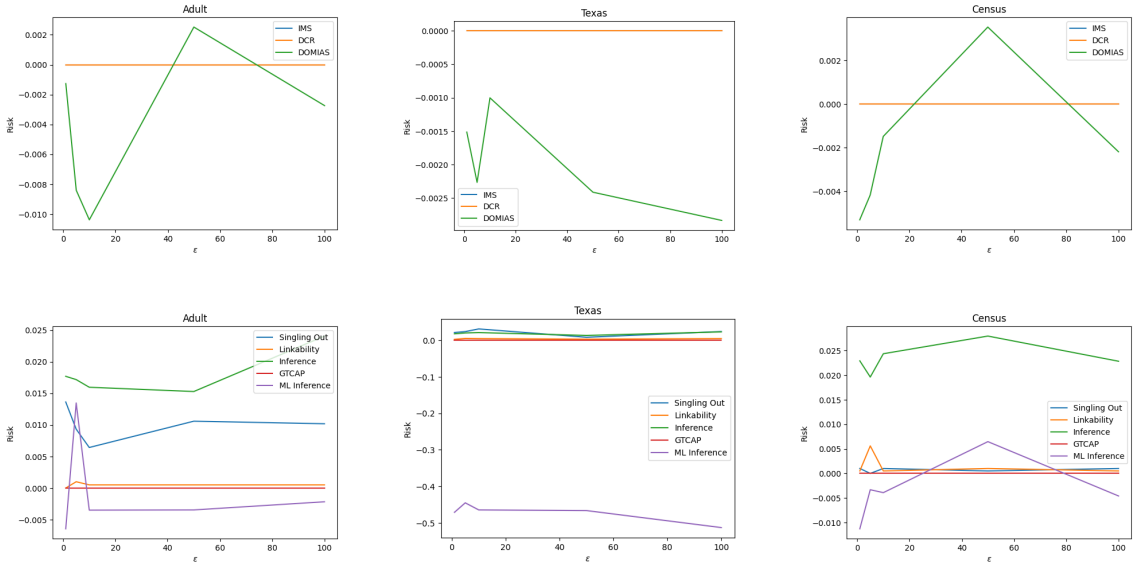


Figure A.4: Risk assessment methods evaluated using the DP risk model - PATEGAN [69]

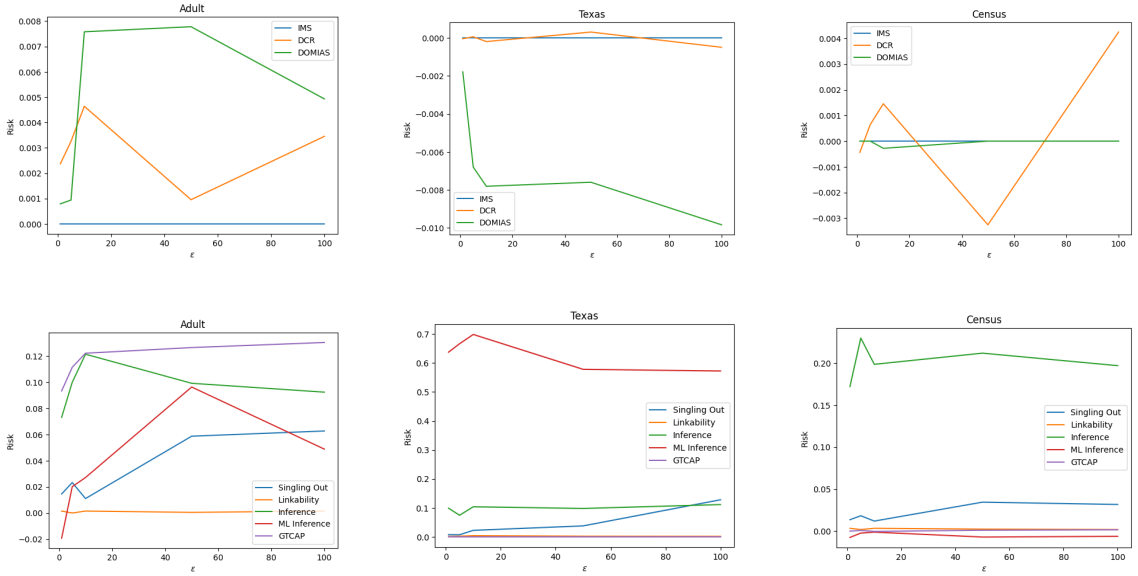


Figure A.5: Risk assessment methods evaluated using the DP risk model - AIM [89]

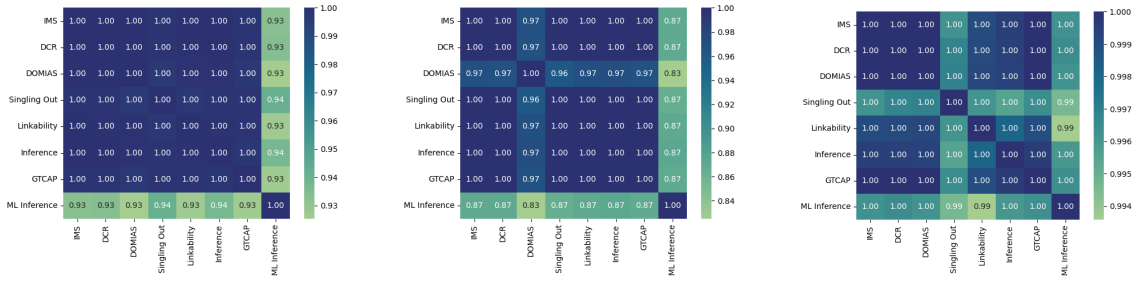


Figure A.6: correlation matrices risk assessment methods using leaky risk model (from left to right: Adult, Texas, Census dataset)

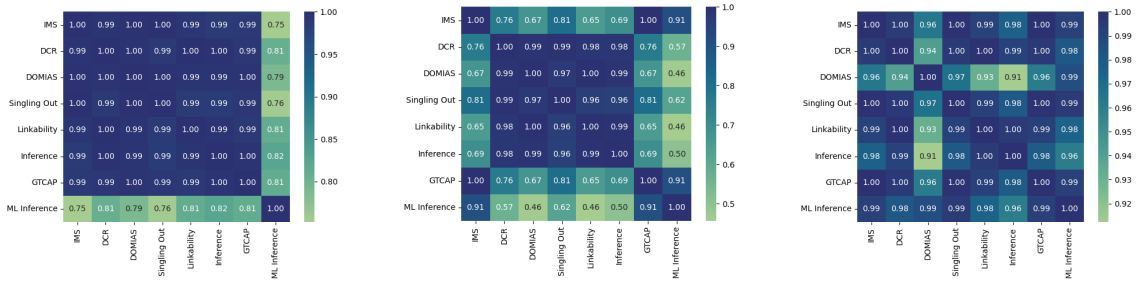


Figure A.7: correlation matrices risk assessment methods using overfitting risk model - RTF [122] left to right: Adult, Texas, Census dataset)

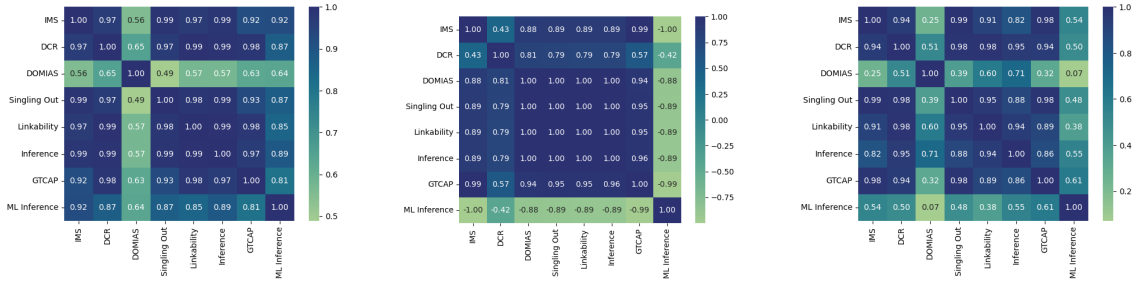


Figure A.8: correlation matrices risk assessment methods using overfitting risk model - Synthpop [95](from left to right: Adult, Texas, Census dataset)

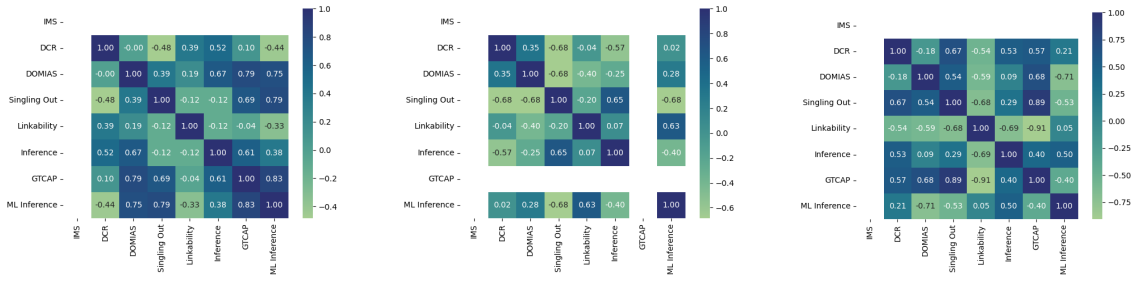


Figure A.9: correlation matrices risk assessment methods using DP risk model - AIM [89] (from left to right: Adult, Texas, Census dataset)

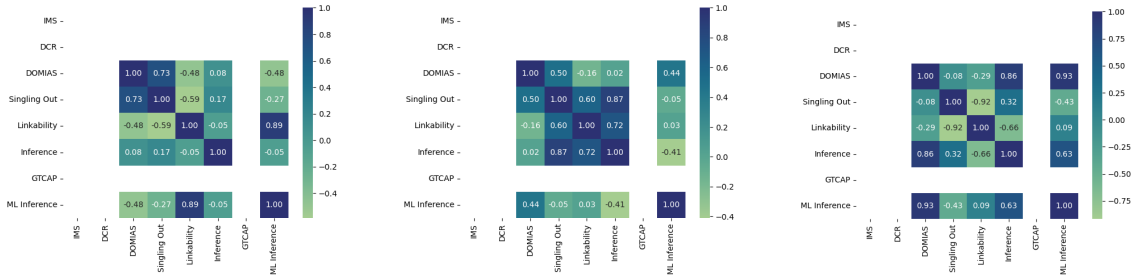


Figure A.10: correlation matrices risk assessment methods using DP risk model - PATE-GAN [69] (from left to right: Adult, Texas, Census dataset)

A.2 Experiments with k-NN-based indicators

The study in this section focused on generalizing the Distance to Closest Record (DCR) by examining the k-nearest neighbors instead of just the single closest one 4.5. The goal was to determine if considering a wider local neighborhood provides more robust or sensitive privacy risk estimates. Across all datasets and risk models, the indicator based on $k = 1$ consistently outperformed higher values of k. The research concluded that examine the immediate neighborhood is the most effective way to detect localized memorization or direct leaks. Larger k values tended to smooth out or "dilute" the signal of these specific vulnerabilities, making the metric less sensitive to individual record disclosure.

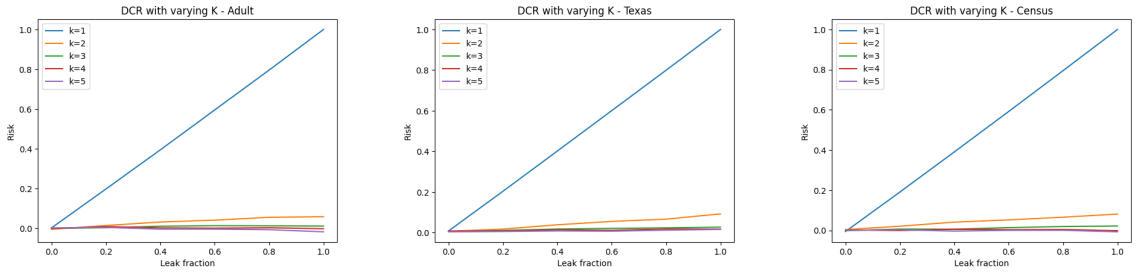


Figure A.11: Results with k-NN-based privacy metric for various k using the leaky risk model

A.3 Experiments with outlier removal

This section details experiments aimed at understanding the relationship between vulnerable outliers and overall privacy risk 4.5. We used the local outlier factor (LOF) to embeddings obtained through contrastive learning to identify outliers [99]. Different proportions of outliers were removed to evaluate their impact on the measurements. We used the overfitting risk model, with overfit ratios of 1.0 and 1.6.

For each metric, we applied bootstrap sampling with replacement from the original dataset ($1,000$ resamples per configuration apart from MIA and GTCAP where we resample 10 times). Metrics were recomputed for each resample, allowing us to estimate confidence intervals. The resulting error bands correspond to 95% confidence intervals.

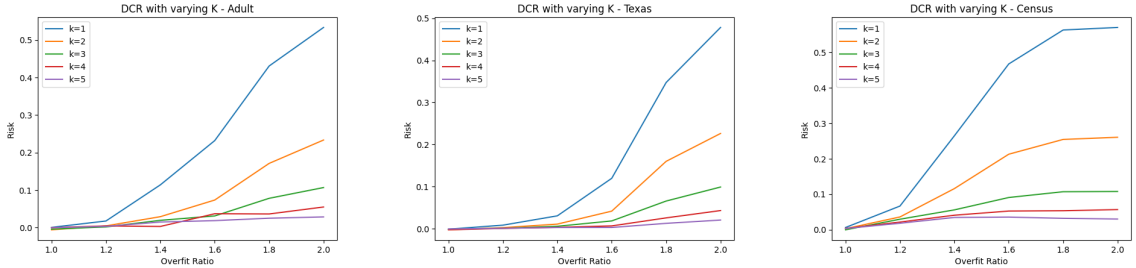


Figure A.12: Results with k-NN-based privacy metric for various k using the overfit risk model - RTF [122]

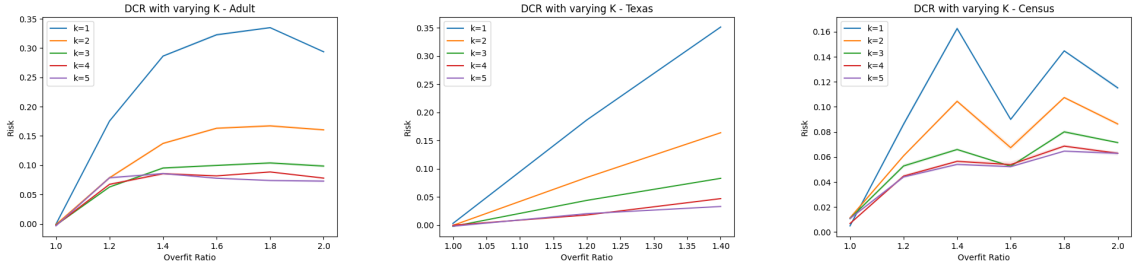


Figure A.13: Results with k-NN-based privacy metric for various k using the overfit risk model - Synthpop [95]

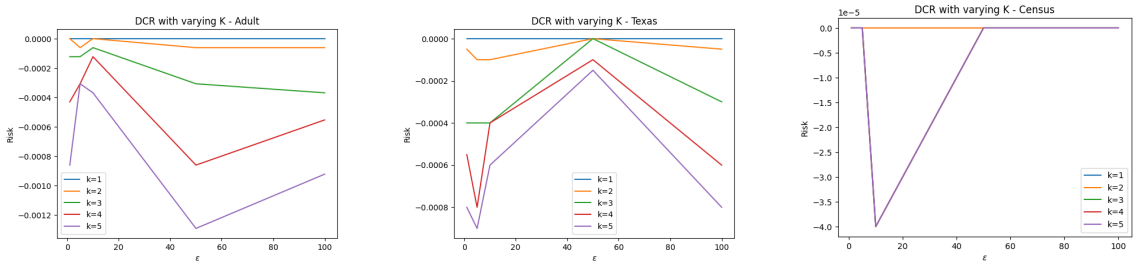


Figure A.14: Results with k-NN-based privacy metric for various k using the DP risk model - PATEGAN [69]

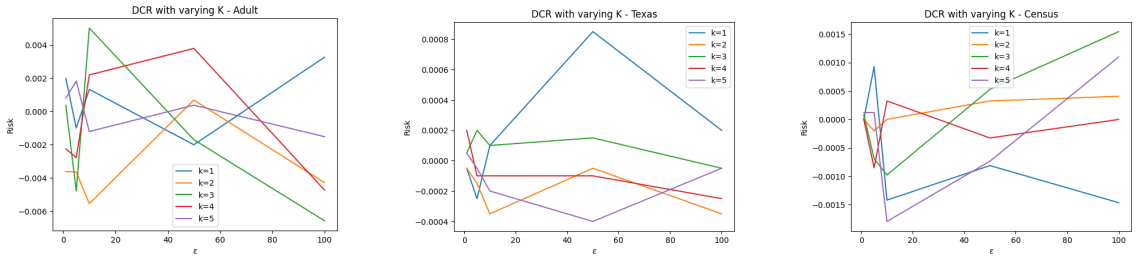


Figure A.15: Results with k-NN-based privacy metric for various k using the DP risk model - AIM[89]

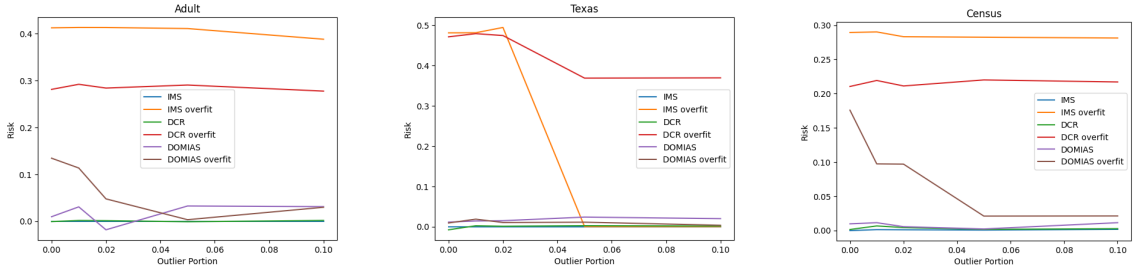


Figure A.16: IMS, DCR, and MIA with outlier removal in the original dataset prior to generator training, with both no overfitting ($f_o = 1$) and overfitting ($f_o = 1.6$) - RTF [122]

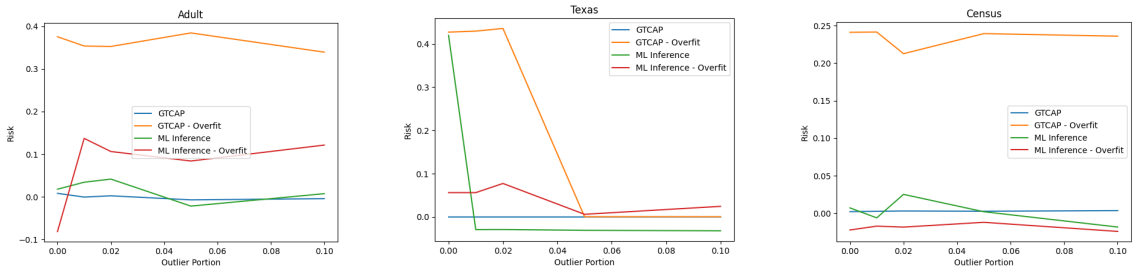


Figure A.17: GTCAP and ML Inference with outlier removal in the original dataset prior to generator training, with both no overfitting ($f_o = 1$) and overfitting ($f_o = 1.6$) - RTF [122]

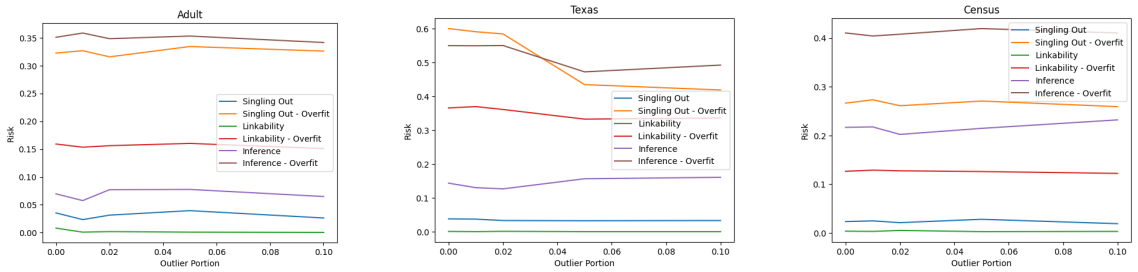


Figure A.18: Anonymeter's methods with outlier removal in the original dataset prior to generator training, with both no overfitting ($f_o = 1$) and overfitting ($f_o = 1.6$) - RTF [122]

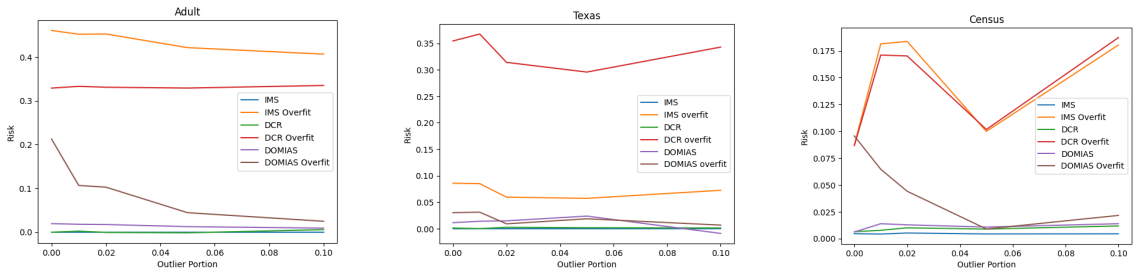


Figure A.19: IMS, DCR, and MIA with outlier removal in the original dataset prior to generator training, with both no overfitting ($f_o = 1$) and overfitting ($f_o = 1.6$) - Synthpop [95]

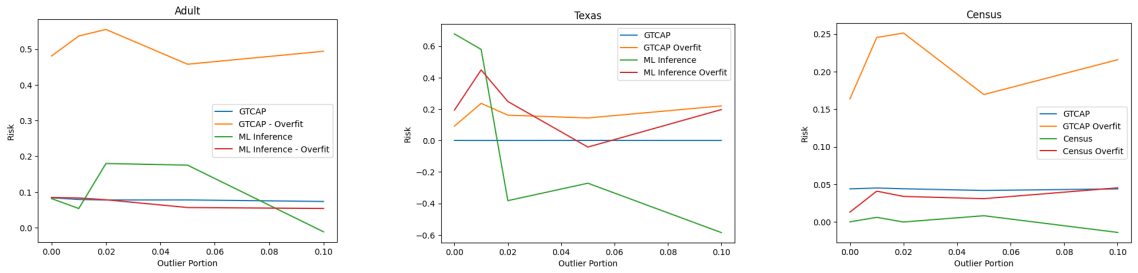


Figure A.20: GTCAP and ML Inference with outlier removal in the original dataset prior to generator training, with both no overfitting ($f_o = 1$) and overfitting ($f_o = 1.6$) - Synthpop [95]

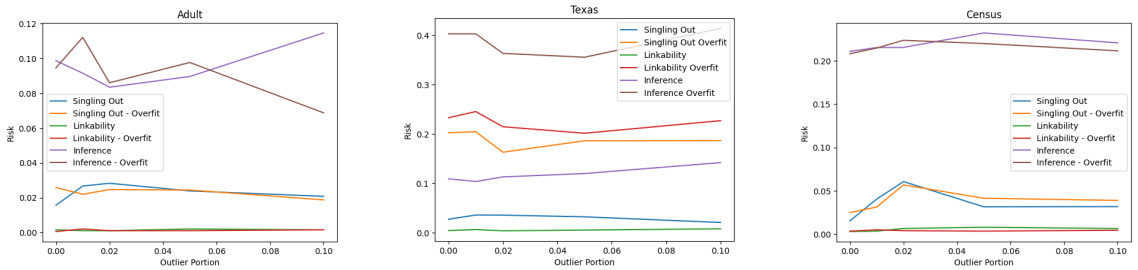


Figure A.21: Anonymeter's methods with outlier removal in the original dataset prior to generator training, with both no overfitting ($f_o = 1$) and overfitting ($f_o = 1.6$) - Synthpop [95]

However, we observed that the variance across bootstrap samples is very low, typically on the order of 10^{-3}. As a result, we decided not to plot the confidence intervals, as doing so would add visual clutter without contributing to the interpretability of the figures.

A.4 Hybrid Pipeline Results

We present four key case studies to illustrate our findings. First, we examine the configuration that achieved the optimal balance considering Singling Out, Inference and ML Efficacy. Following this, we isolate the specific results that yielded the highest performance improvements for each metric individually.

The Best Overall Configuration

To achieve the best overall balance across all evaluation criteria, the pipeline was configured using k-anonymity with $k = 5$ and a Post-Uniform transition method. The data underwent stratified sampling with a sampling fraction of $\beta = 1$, while CTGAN served as the synthetic data generator.

Metric	Singling Out	Inference	ML Efficacy	TVD	WD
	48.3%	52.1%	−8.1%	27.8%	−204.4%

Table A.1: Performance metrics for the optimal combined configuration.

The results indicate a strong privacy-utility trade-off, where effective risk mitigation in Singling Out and Inference is achieved at the cost of a minor reduction in ML Efficacy. However, the significant divergence in Wasserstein Distance suggests that the synthetic generation process notably alters the original data distribution.

Singling Out

To achieve the best results considering Singling Out only, the pipeline was configured using k-anonymity with $k = 20$ and a Post-Uniform transition method. The data underwent stratified sampling with a sampling fraction of $\beta = 1$, while RTF served as the synthetic data generator.

Metric	Singling Out	Inference	ML Efficacy	TVD	WD
	61.5%	6.7%	−42.6%	−16.5%	−412.6%

Table A.2: Results for the configuration optimized for Singling Out.

This configuration achieved the highest Singling Out score among all experiments, providing the most robust defense against individual record identification. However, this peak privacy performance comes at a significant cost to utility, as evidenced by the sharpest declines in ML Efficacy and Wasserstein Distance across the entire study.

Inference

To achieve the best results considering Inference only, the pipeline was configured using k-anonymity with $k = 20$ and a Pre-Uniform transition method. The data underwent random sampling with a sampling fraction of $\beta = 0.5$, while CTGAN served as the synthetic data generator.

Metric	Singling Out	Inference	ML Efficacy	TVD	WD
	18.8%	64.6%	−17.2%	16.0%	−293.3%

Table A.3: Results for the configuration optimized for Inference.

This configuration achieves excellent privacy protection against Singling Out, though it incurs a more pronounced penalty in ML Efficacy and distributional similarity. The high Inference score alongside a significant Wasserstein Distance suggests that while individual records are well-hidden, the overall statistical structure is heavily modified.

ML Efficacy

To achieve the best results considering ML Efficacy only, the pipeline was configured using k-anonymity with $k = 10$ and a Pre-Conditional transition method. The data underwent stratified sampling with a sampling fraction of $\beta = 1$, while Tabddpm served as the synthetic data generator.

Metric	Singling Out	Inference	ML Efficacy	TVD	WD
	−42.8%	−38.8%	7.9%	33.1%	46.9%

Table A.4: Results for the configuration optimized for ML Efficacy.

This configuration yielded the highest ML Efficacy observed across all tests, indicating that the synthetic data effectively retains the predictive power of the original dataset. While the privacy metrics show increased vulnerability, the positive gains in both ML Efficacy and Wasserstein Distance demonstrate superior maintenance of the data's statistical and geometric properties.

UNIVERSITÀ
DEGLI STUDI
DI TRIESTE

La borsa di dottorato è cofinanziata con risorse dell'Unione europea, NextGeneration EU - Piano Nazionale di Ripresa e Resilienza, Missione 4 – Componente 2– Investimento 3.3 CUP J92B22001000007.