Summary

Synthetic data generators (SDGs) as a means of preserving data confidentiality is an active field of research. These techniques are in high demand due to the many use cases where they can be employed. For example, within the medical field, there is an increasing demand for generating synthetic electronic health records (EHRs), it is showcased during the COVID-19 pandemic where there was a rapid demand need for healthcare solutions.

In recent years, various SDGs have been proposed. These generally work by injecting noise into the data. Many of the SDGs are built to be differentially private (DP). One such method is PrivBayes, which utilises ε-differential privacy (ε-DP) and Bayesian networks to generate synthetic data. Here, ε-DP is a property that allows researchers to reason about the amount of noise necessary to inject into the data during generation to obtain private data.

The goal of differential privacy (DP) is to protect sensitive information that is contained in the single individual's data. However, depending on the goal of the adversary, DP may not be sufficient.

Within the medical field, where all data is considered sensitive, there is concern that an individual's attributes may not be private given that an adversary has additional information, which might help the adversary deduce the sensitive attribute values of said individual. This concern is induced by the utility privacy trade-off, which deduces that any data with utility is at risk against privacy attacks. Therefore, a more comprehensive estimation of data privacy requires the development of other privacy metrics.

DP algorithms do not ensure that nothing about an individual can be learnt from the synthetic data, as this can only be achieved by releasing no information. This means that if the synthetic data has utility, it enables adversaries to extract sensitive information about individuals regardless of their data being included, as there is a clear trade-off between utility and privacy in synthetic data. SDGs such as PrivBayes try to balance this trade-off by using ε-DP. It has, however, been shown that these methods are subject to the same trade-off, while making it impossible to predict which data features are preserved in the synthetic data.

Since utility takes priority over privacy, the focus for many researchers is on utility. Current privacy metrics often use frequentist statistics for their estimation. A Bayesian approach would, however, allow us to model the problem in different scenarios, e.g., different knowledge being available to an adversary. Due to this, Reiter et al. attempts to address these issues by formulating a Bayesian estimation of the Attribute Disclosure Risk (ADR) for binary variables (i.e., the probability that an adversary discloses sensitive values of the synthesised attributes on an individual) in synthetic data, where an adversary can use the released synthetic data and any other information to infer the attributes of an individual. Hornby & Hu implemented a proof of concept of Reiter et al.'s Bayesian estimation of ADR for a mixture of discrete and continuous variables. Hornby & Hu's focus were on implementing these concepts as a demonstration, but with little regards to practical applicability, like estimating the risk of attribute disclosure in synthetic data generated by, e.g., a Bayesian DP-SDG such as PrivBayes. Therefore, in this project, we studied the following problem:

Problem statement *How do we compute the ADR of continuous synthetic data generated by PrivBayes?*

To address this problem we presented a coupling between PrivBayes and Hornby & Hu where we tested this coupling in two experiments. Experiment 1, investigate the correlation between ε and the chance of correctly identifying an attribute value, and Experiment 2 investigate the effect of an injected outlier, who has extreme values inside and outside the range of values in the dataset. Here Hornby & Hu, seemed promising due to its ability to measure the risk of attribute disclosure

given, the synthetic data, real data, auxiliary information an adversary might know, and information about the synthetic data generation, with the latter two inputs being relatively novel, considering the domain of private synthetic data. However, despite this extra information, they only demonstrate minor changes to the chance of guessing continuous attribute values correctly in their testing. This is something we tested more extensively with PrivBayes as the synthesiser, and our results, like theirs, show little change in ADR for continuous attributes, which is unexpected considering the additional information available to the adversary. Furthermore, the results also demonstrated that the ADR for continuous attributes was not directly influenced by the amount of noise injected by PrivBayes. This could indicate that this additional information might be insignificant or distracting when using PrivBayes, making guessing correctly harder. Despite this, we believe that Bayesian modelling provides an estimation where we are able to adjust the knowledge available to the adversary, which can provide more accurate results in the vision of protecting individuals' sensitive attributes.

Bayesian Estimation of Attribute Disclosure Risk of PrivBayes

FREDERIK MARINUS TRUDSLEV, Department of Computer Science, Aalborg University, Denmark
SILAS OLIVER TORUP BACHMANN, Department of Computer Science, Aalborg University, Denmark

Synthesizing is becoming ever more important as a means of providing anonymous data. However, guaranteeing the anonymity of individuals using synthetic data is still an open problem, which is reflected by the number of synthetic data generators (SDG) and privacy metrics that have been proposed in recent years. One mathematical framework that is often used to ensure privacy of SGDs is differential privacy (DP) which guarantees that adding or removing any individual from a real dataset does not significantly change the distribution. However, the guarantees provided by DP causes concern in fields where all attribute values must remain secret, as attacks that attempts to guess an unknown attribute value of a given individual are prevalent. This reinforces the need for privacy metrics that models this type of attack. Despite this, many privacy metrics mostly only consider an adversary's knowledge of the synthetic dataset, but in reality an adversary might have knowledge beyond the synthetic data, such as how the synthetic data was generated or knowledge of an individual not included in the real dataset. A reason for this might be the use of frequentist statistics rather than Bayesian statistics, where the latter provides the ability to continuously update one's beliefs with new information, making it a more natural fit for modelling of different degrees of adversary knowledge. Hornby & Hu has implemented a Bayesian variation of calculating the risk of attribute inference attack, which accounts for an adversary's auxiliary knowledge as well as knowledge about the synthesisation method used. From this, we propose an implementation that couples PrivBayes, a differentially private Bayesian SDG, to Hornby & Hu with the purpose of investigating Hornby & Hu's ability to assess the risk of disclosing continuous attribute values for a synthetic dataset generated by the DP method PrivBayes given two different scenarios with different datasets. One, where we use different ε-values for DP and another, where we inject outliers into the real dataset. Despite the extra information, our results showed low risk of disclosing continuous attributes for all ε-values. Furthermore, the results also demonstrated that the attribute disclosure risk for continuous attributes was not directly influenced by the amount of noise injected by PrivBayes. Despite this, we believe that Bayesian modelling provides an estimation where we are able to adjust the knowledge available to the adversary, which can provide more accurate results in the vision of protecting individuals' sensitive attributes.

1 INTRODUCTION

Synthetic data generators (SDGs) as a means of preserving data confidentiality is an active field of research [1][2][3]. These techniques are in high demand due to the many use cases where they can be employed. For example, within the medical field, there is an increasing demand for generating synthetic electronic health records (EHRs), it is showcased during the COVID-19 pandemic where there was a rapid demand need for healthcare solutions [4].

In recent years, various SDGs have been proposed. These generally work by injecting noise into the data. Many of the SDGs are built to be differentially private (DP). One such method is PrivBayes [3], which utilises ε-differential privacy (ε-DP) and Bayesian networks to generate synthetic data. Here, ε-DP is a property that allows researchers to reason about the amount of noise necessary to inject into the data during generation to obtain private data.

The goal of differential privacy (DP) is to protect sensitive information that is contained in the single individual's data. However, depending on the goal of the adversary, DP may not be sufficient [5][6].

Within the medical field, where all data is considered sensitive, there is concern that an individual's attributes may not be private given that an adversary has additional information, which might help the adversary deduce the sensitive attribute values of said individual. This concern is induced by the utility privacy trade-off, which deduces that any data

Authors' addresses: Frederik Marinus Trudslev, Department of Computer Science, Aalborg University, Aalborg Ø, Denmark, ftruds19@student.aau.dk; Silas Oliver Torup Bachmann, Department of Computer Science, Aalborg University, Aalborg Ø, Denmark, sbachm19@student.aau.dk.

with utility is at risk against privacy attacks [7]. Therefore, a more comprehensive estimation of data privacy requires the development of other privacy metrics.

DP algorithms do not ensure that nothing about an individual can be learnt from the synthetic data, as this can only be achieved by releasing no information [8]. This means that if the synthetic data has utility, it enables adversaries to extract sensitive information about individuals regardless of their data being included, as there is a clear trade-off between utility and privacy in synthetic data [7]. SDGs such as PrivBayes try to balance this trade-off by using ε-DP. It has, however, been shown that these methods are subject to the same trade-off, while making it impossible to predict which data features are preserved in the synthetic data [7].

Since utility takes priority over privacy, the focus for many researchers is on utility [9]. Current privacy metrics often use frequentist statistics for their estimation, as elaborated in Section 2. A Bayesian approach would, however, allow us to model the problem in different scenarios, e.g., different knowledge being available to an adversary. Due to this, Reiter et al. [10] attempts to address these issues by formulating a Bayesian estimation of the Attribute Disclosure Risk (ADR) for binary variables (i.e., the probability that an adversary discloses sensitive values of the synthesised attributes on an individual) in synthetic data, where an adversary can use the released synthetic data and any other information to infer the attributes of an individual. Hornby & Hu [11] implemented a proof of concept of Reiter et al.'s Bayesian estimation of ADR for a mixture of discrete and continuous variables. Hornby & Hu's focus were on implementing these concepts as a demonstration, but with little regards to practical applicability, like estimating the risk of attribute disclosure in synthetic data generated by, e.g., a Bayesian DP-SDG such as PrivBayes. Therefore, in this project, we study the following problem:

Problem statement *How do we compute the ADR of continuous synthetic data generated by PrivBayes?*

To address this, we propose a coupling between PrivBayes and Hornby & Hu. Here, PrivBayes is chosen as it also uses a Bayesian approach, which makes the coupling simpler. Continuous variables were chosen, as they can represent values of any range, thereby generally representing a greater granularity than discrete variables, making it a more interesting target for guessing attribute values. We then test this coupling in two experiments. Experiment 1, investigate the correlation between ε and the chance of correctly identifying an attribute value, and Experiment 2 investigate the effect of an injected outlier, who has extreme values inside and outside the range of values in the dataset. Here, the results show no correlation between *varepsilon* and the chance of identifying an attribute value with or without outliers, except for categorical values which showed surprisingly good results.

2 RELATED WORK

In this Section, we investigate how the state-of-the-art model attribute inference attacks, with the goal of investigating whether they use Bayesian statistics and what knowledge an adversary may know beyond the synthetic dataset.

Stadler et al. [7] present a frequentist privacy evaluation of DP-SGDs, where they quantify how much advantage the different SGDs give to an adversary using linkability/membership inference, and attribute inference as attack vectors. Here, the adversary uses machine learning along with knowledge of synthetic data to conduct the attack. Here. the adversary does not have any additional information. They conclude that DP, when implemented correctly, reduces the privacy gain of these two attacks, but at the same utility cost as traditional row level sanitization.

Zhang et al. [12] introduce the Synthetic Data Vault (SDV) framework for generating and testing synthetic data. They include multiple models for synthetic data generation and metrics like CategoricalCAP, a type of attribute inference risk estimation. CategoricalCAP does this by combining public and synthetic information, with public knowledge being a subset of the real dataset. CategoricalCAP then counts the instances of values for the same attribute as the unknown attribute value, where the individuals are identical to the individual that has the attribute we want to guess. It then gives a probabilistic score based on the frequency of attribute values as a guess of the unknown attribute value. While CategoricalCAP does consider auxiliary information, it is not suited for continuous variables, as it relies on repeated attribute values for informed guessing.

Mehnaz et al. [13] propose multiple model attacks on classifiers, where each model attack assumes different adversary knowledge about the data and classifier for these attacks, e.g., knowledge of the classifier's confusion matrix or predicted labels. Here, the simplest attack assumes that the adversary have knowledge of the probability distributions and guesses the value with the highest marginal prior. This is quite similar to how Reiter and Hornby & Hu model their attack. Mehnaz et al. However are not truly Bayesian as they only rely on their prior and does not attempt to update it.

Hernandez et al. [14] introduce a synthetic tabular data evaluation pipeline in three dimensions (resemblance, utility, and privacy); here, privacy consists of distance and inference risk metrics, where they train a machine learning model on the synthetic dataset and try to predict the unknown values of sensitive attributes of the real dataset. Like other solutions presented, Hernandez et al. does not model an adversary with knowledge of the synthetic data generation process and, likewise, do not use Bayesian statistics.

In summary, we see that some of the presented privacy attacks, model an adversary with more information than just the synthetic dataset. Here, Zhang et al. [5] model a subset of the real dataset that an adversary might have knowledge about, and Hernandez et al. [14] models an adversary, which has access to a subset of the real dataset. Which is contrary to Reiter et al. [10] and Hornby et al. [11] which uses Bayesian statistics to model an adversary that has information about the synthesisation model and information about the real dataset.

3 PROBLEM DEFINITION

In this Section, we formally define the problem of estimating the ADR of continuous synthetic data generated by PrivBayes. First, we explain a Bayesian approach to private synthetic data generation. From this, we highlight one of the current problems with DP algorithms, after which, we elaborate on our contribution.

3.1 Private Synthetic Data Generation

Let $y_i = (y_{i1}, \ldots, y_{ih})$ be a vector of information about an individual i in the confidential real dataset. The entire real dataset is thereby denoted as $y = \{y_1, \ldots, y_n\}$. From the real dataset, a synthetic dataset can be generated. The synthetic dataset generation can be modelled as a Bayesian hierarchical model, illustrated by the Directed Acyclic Graph (DAG) in Figure 1.

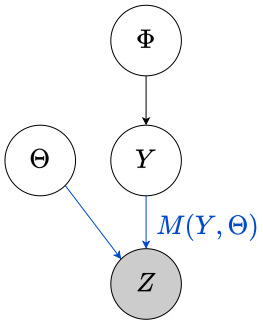

Fig. 1. DAG representing a Bayesian hierarchical model of synthetic data generation. For the representation of Bayesian hierarchical models, we use the notation used by Christopher M. Bishop [15] where nodes are stochastic variables. Nodes with a grey fill are the observable variables. In this case, observable refers to the available knowledge of an adversary. Furthermore, the directed edges describe the causal relation present in the model.

Here, Y is a stochastic variable such that $Y \mid \Phi \sim F$, where Φ defines the parameters of some distribution F. From Y, the real dataset y of individuals is sampled. To generate a synthetic dataset z, a SDG M is applied with the parameters Y and Θ, where Θ represent the parameters used in synthesising, such as ε for differential privacy. Thereby, we have that $Z = M(Y, \Theta)$ where Z is a stochastic variable, meaning that $Z \mid Y = y \sim G$ and $M(Y, \Theta) \mid Y = y \sim G$ for some distribution G.

3.2 Differential Privacy

A way to protect all individuals in a real dataset y against sensitive information leakage is to employ a method M compliant with differential privacy. For M to be differentially private, the following must hold:

Definition 1. *(ε-Differential Privacy [16]). A synthetisation method M is ε-differentially private if:*

(1) $\forall$ *real datasets* $y_{(1)}, y_{(2)}$ *that differ by one individual, such that* $|y_{(1)} \backslash y_{(2)}| = 1 \vee |y_{(2)} \backslash y_{(1)}| = 1$, *and*

(2) $\forall\, o \subset O$, *with* O *being the set of possible outputs, we have that:*

$$\frac{P(M(y_{(1)}, \theta) \in o)}{P(M(y_{(2)}, \theta) \in o)} \leq e^{\varepsilon} \tag{1}$$

Here, $P(M(y_{(1)}, \theta) \in o)$ is the probability of an output in the set o using method M with parameters θ on dataset $y_{(1)}$, in other words. *"Changing a single individual's data in the database leads to a small change in the distribution of the outputs"* [17]. ε-differential privacy therefore gives a strong guarantee of confidentiality, as an adversary cannot learn more about y_i given that the adversary has knowledge about $y \setminus \{y_i\}$, the SDG M and the synthetic dataset z [18].

3.3 ADR of Synthetic Data

Due to the lack of guarantee from differential privacy about the adversary's chance of disclosing a specific individual's attributes, Reiter et al. [10] proposed a framework for Bayesian estimation of ADR in synthetic data. Using their framework, we want to know what an adversary can learn about individuals' attribute given that the adversary has knowledge about z, M and Θ. The problem then becomes to estimate the density p of the individuals $y_i \in y \forall i \in [1, \ldots, n]$ given the values of z, θ and ϕ:

$$\begin{aligned} p(y \mid z, \theta, \phi) &= \frac{p(y, z, \theta, \phi)}{p(z, \theta, \phi)} = \frac{p(z \mid y, \theta, \phi) p(y, \theta, \phi)}{p(z, \theta, \phi)} \\ &= \frac{p(z \mid y) p(y \mid \theta, \phi) p(\theta) p(\phi)}{p(z, \theta, \phi)} \\ &= \frac{p(z \mid y) p(y \mid \theta, \phi) p(\theta) p(\phi)}{\int p(z, y, \theta, \phi) dy} \\ &= \frac{p(z \mid y) p(y \mid \theta, \phi)}{\int p(z \mid y, \theta, \phi) p(y \mid \theta, \phi) dy} \end{aligned} \tag{2}$$

Here, y, z, θ and ϕ represent the realised values of Y, Z, Θ and Φ from the DAG in Figure 1. The posterior distribution $p(y \mid z, \theta, \phi)$ is the density of what we can learn about individuals in the real data given some observations. This can be calculated using the likelihood $p(z \mid y)$, which is the probability of our synthetic data given the real individual or the likelihood of y given z multiplied by our prior belief of y, $p(y \mid \theta, \phi)$, and then dividing it by the normalising constant $\int p(z \mid y, \theta, \phi) p(y \mid \theta, \phi) dy$. An example of how the ADR is calculated for continuous data can be found in Section 4.1.

3.4 Linear Regression Synthesizer

In the framework proposed by Hornby and Hu [11], they also provide a method for synthesis. This method is a sequential Bayesian synthesiser using linear regression, where if we have three attributes, a synthetic individual in the dataset is a sample from a probability distribution such that:

$$z_i \sim p(y_{i1}, y_{i2}, y_{i3}) = p(y_{i1}) p(y_{i2} \mid y_{i1}) p(y_{i3} \mid yi1, yi2) \tag{3}$$

Here, synthesis is performed sequentially, meaning that the synthetic attributes are sampled individually. For this example, first a linear regression model would be fitted on the real values $\{y_{i1} \mid i \in [1 \ldots n]\}$ with a constant as predictor to get the parameters Θ that describe the distribution. Synthetic values for the first attribute, $\{z_{i1} \mid i \in [1 \ldots m]\}$, are then sampled from the posterior predictive distribution using a Monte Carlo approach. Second, a linear regression model is fitted on the real values $\{y_{i2} \mid i \in [1 \ldots n]\}$ with $\{y_{i1} \mid i \in [1 \ldots n]\}$ as a predictor, where synthetic data for the second attribute are sampled from the posterior predictive distribution using the parameters learnt from fitting a model as well as the synthetic values ($\{z_{i2} \mid i \in [1 \ldots m]\}$). Third, the same approach is used to generate synthetic samples for the third attribute from the posterior predictive distribution using the synthetic values ($\{z_{i1} \mid i \in [1 \ldots m]\}$ and $\{z_{i2} \mid i \in [1 \ldots m]\}$) and parameters learnt by fitting a linear model on the real values $\{y_{i3} \mid i \in [1 \ldots n]\}$ with

$\{y_{i1} \mid i \in [1 \dots n]\}$ and $\{y_{i2} \mid i \in [1 \dots n]\}$ as predictor variables. An example of this method for synthesis using continuous attributes can be found in Section 4.1.

3.5 The Problem of Applicability

The linear regression synthesiser proposed by Hornby and Hu [11] provides a solution that allows them to have great control of both the method M and the parameters Θ used for synthesis, which is very useful when calculating the ADR, as all the necessary parameters of equation 2 are given. This is due to them being able to directly ascertain these from the linear regression model, as this directly gives the parameters that are used for synthesis. Thereby, their method actually has two outputs, such that the linear regression synthesiser provides them with both Θ and Z. However, as elaborated in Section 3.1, a SDG is a typical method such that $Z = M(Y, \Theta)$. Therefore, due to the linear regression synthesiser being tailored to give necessary parameters for risk measurement, while other SDGs do not, the problem of applicability arises when we want to measure the ADR of synthetic data generated by any other SDG.

4 BACKGROUND

This Section covers the essential methods used to calculate the ADR of continuous synthetic data generated by the DP method, PrivBayes. First, we present the method for estimating ADR for continuous synthetic data. Second, we present how continuous synthetic data is generated using PrivBayes.

4.1 ADR of Continuous Synthetic data

This Section presents a running example of measuring the ADR of a synthesized dataset obtained by sampling individuals from a joint distribution over h attributes, an individual in the real dataset is on the form $(y_{i1}, y_{i2}, \ldots, y_{ih})$. There exists a network $\mathcal{B}$ that describes the causal relations between these variables, such that an individual in the real dataset is sampled from a probability distribution in the following way:

$$y_i \sim p(y_{i1}, y_{i2}, \ldots, y_{ih}) = \prod_{j=1}^{h} p(y_{ij} \mid \Pi_j), \tag{4}$$

where Π_j is the parent set of attribute y_{ij}.

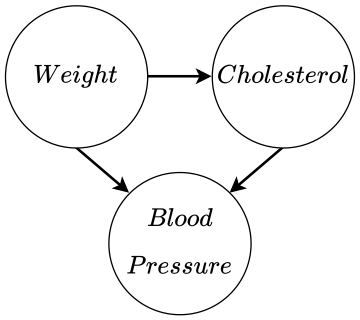

Fig. 2. A Bayesian network representing the causal relations between attributes ($y_{i1}\ \forall\ i \in [1 \ldots n]$ is the individuals' *weight*, $y_{i2}\ \forall\ i \in [1 \ldots n]$ is the individuals' *cholesterol*, $y_{i3}\ \forall\ i \in [1 \ldots n]$ is the individuals' *blood pressure*) in the real dataset. It can be seen that *weight* has a causal effect on *cholesterol* and *blood pressure*, while *cholesterol* has a causal effect on *blood pressure*.

Example 4.1. Let $h = 3$, where $y_{i1}\ \forall\, i \in [1 \ldots n]$ is the *weight* attribute values, $y_{i2}\ \forall\, i \in [1 \ldots n]$ is the *cholesterol level* attribute values and $y_{i3}\ \forall\ i \in [1 \ldots n]$ is the *blood pressure* attribute values. Furthermore, we have an individual y_{Bob}, that has attribute values $(105.3, 232.34, 130.3)$. The parent set contains all parent attributes such that in Figure 2, we have, e.g., $\Pi_3 = \{weight, cholesterol\}$. The causal relations present in this example of a real dataset are shown in Figure 2.

Since there are causal relations between attributes, we can make assumptions about the distributions from which these are sampled. Hornby and Hu [11] assume that the attributes are sampled from a normal distribution, with the causal relation between attributes being a linear relation. We can, therefore, fit a generalised linear model on the real dataset to estimate parameters that capture the distribution of attributes and the linear relations between them. By fitting a generalised linear model on the real dataset, we derive a set of parameters $\theta = (\mu, \sigma_1, \alpha_1, \beta_1, \sigma_2, \alpha_2, \beta_2, \beta_3, \sigma_3)$

that estimate the conditional distributions of individuals in the real dataset of our example. These parameters are sampled from distributions that lead to the predicted values being close to the true values of y_{ij}. Prior to fitting the generalised linear model, the parameters are sampled as:

$$\begin{aligned} \mu, \alpha_k &\sim_{approx} \phi\left(\frac{1}{N}\sum_{i=1}^{N} y_{ij}, 2.5\right), \\ \sigma_j &\sim_{approx} Exp(\lambda = 2), \\ \beta_k &\sim \phi(0, 2), \end{aligned} \tag{5}$$

where $\sim_{approx}$ is the prior approximation of the distribution, which is adjusted by fitting the model on the real dataset. To generate synthetic data, Hornby and Hu [11] propose a sequential Bayesian synthesiser. Here, they estimate Θ using the generalised linear model, and sample synthetic data from the model that best capture the distribution of the attributes in the data such that:

$$\begin{aligned} z_{i1} &\sim \phi(\mu, \sigma_1) \\ z_{i2} \mid y_{i1} &\sim \phi(\alpha_1 + \beta_1 \cdot y_{i1}, \sigma_2) \\ z_{i3} \mid y_{i1}, y_{i2} &\sim \phi(\alpha_2 + \beta_2 \cdot y_{i1} + \beta_3 \cdot y_{i2}, \sigma_3) \end{aligned} \tag{6}$$

Suppose we have a real dataset y that contains $i = 1, \ldots, n = 1000$ individuals (y_i) such that $y = \{y_1, \ldots, y_{1000}\}$ with attributes having the same causal relations as shown in Figure 2. We then generate a synthetic dataset $z = \{z_1, \ldots, z_m\}$ where $m = 1000$.

To illustrate the ADR measurement, we assume the adversary's target is $y_{1000} = y_{Bob}$, and a worst-case scenario, where the adversary's auxiliary information A consists of knowledge about y_i for all individuals in the real dataset except y_{Bob}, such that $A = y \setminus \{y_{1000}\}$. Furthermore, the adversary also has knowledge about the synthesis process (S). We can thereby model the knowledge an adversary has available as a Bayesian network, shown in Figure 3.

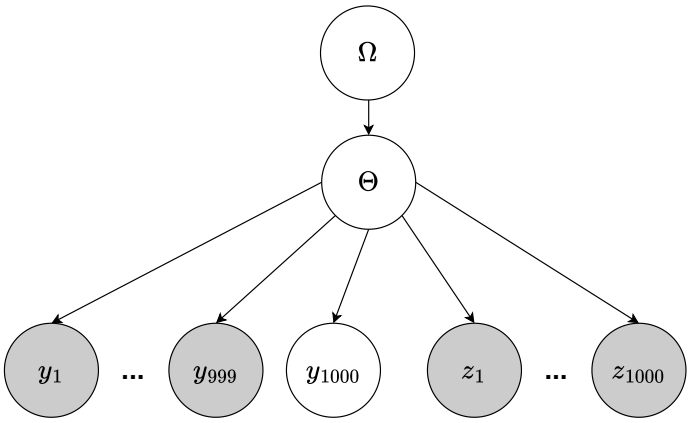

Fig. 3. A Bayesian network representing the causal relations between the real dataset and the synthetic dataset through the parameters in theta. From the Figure, it can be seen that Θ has an effect on both the real and synthetic individuals, while being affected by Ω which is the variance in the estimation of parameters in Θ.

The goal of measuring the ADR is to obtain the posterior probability of a guess $p(Y_{1000} = y^*_{1000} \mid z, A, S)$, where Y_{1000} is a variable which represents an adversary's uncertain knowledge about y_{1000} and y^*_{1000} represents the adversary's guess on the true confidential value of y_{1000}. We can then write the posterior probability in the following form:

$$p(Y_{1000} = y^*_{1000} \mid z, A, S) \propto p(z \mid Y_{1000} = y^*_{1000}, A, S) p(Y_{1000} = y^*_{1000} \mid A, S) \tag{7}$$

Here, Hornby and Hu [11] propose a solution with a uniform prior, such that $p(Y_{1000} = y^*_{1000} \mid A, S) = 1/n$ for all y in the support. By assuming a uniform naive prior, the problem then becomes estimating the likelihood $p(z \mid Y_{1000} = y^*_{1000}, A, S)$, as this is effectively equivalent to an adversary searching over all y^*_{1000} for the highest probability of $p(z \mid Y_{1000} = y^*_{1000}, A, S)$. Here, Monte Carlo approximation is typically used, but as elaborated by Hornby and Hu [11], this is computationally expensive. Therefore, we use importance sampling [19] to estimate the likelihood through a function g of θ, $g(\theta) = p(z \mid Y_{1000} = y^*_{1000}, A, S, \theta)$. To do importance sampling, we introduce the density $f^*(\theta \mid Y_{1000} = y_{1000}, A, S)$, which is a density of the parameter θ given the real data and knowledge about the synthesis process:

$$\begin{aligned}
\mathbf{E}[g(\Theta)] &= \int p(z \mid Y_{1000} = y^*_{1000}, A, S, \theta) f(\theta \mid Y_{1000} = y^*_{1000}, A, S) d\theta \\
&= \int p(z \mid Y_{1000} = y^*_{1000}, A, S, \theta) \frac{f^*(\theta \mid Y_{1000} = y_{1000}, A, S)}{f^*(\theta \mid Y_{1000} = y_{1000}, A, S)} f(\theta \mid Y_{1000} = y^*_{1000}, A, S) d\theta \\
&= \int p(z \mid Y_{1000} = y^*_{1000}, A, S, \theta) \frac{f(\theta \mid Y_{1000} = y^*_{1000}, A, S)}{f^*(\theta \mid Y_{1000} = y_{1000}, A, S)} f^*(\theta \mid Y_{1000} = y_{1000}, A, S) d\theta
\end{aligned} \tag{8}$$

Here, $\frac{f(\cdot)}{f^*(\cdot)}$ is a weighting function that captures the likelihood ratio. The weighting function we use here, however, has a problem, as it is only known up to a proportional relation between the weighting function and the probability we want to measure such that:

$$\frac{f(\cdot)}{f^*(\cdot)} \propto \frac{q(\theta \mid Y_{1000} = y^*_{1000}, A, S)}{q^*(\theta \mid Y_{1000} = y_{1000}, S)} \tag{9}$$

To overcome this proportionality, we use the fact that we can then rewrite the weighting function, where for simplification, the function parameters will be shortened to θ such that:

$$\frac{f(\theta)}{f^*(\theta)} = \frac{\frac{q(\theta)}{\int q(\theta) d\theta}}{\frac{q^*(\theta)}{\int q^*(\theta)} d\theta} = \frac{\frac{q(\theta)}{q^*(\theta)}}{\frac{\int q(\theta) d\theta}{\int q^*(\theta) d\theta}} \tag{10}$$

To estimate taking the integral in the denominators of the equation, we do as previously and introduce an $f^*(\cdot)$ function to both the integrals such that the denominator becomes:

$$\begin{aligned}
\frac{\int q(\theta) d\theta}{\int q^*(\theta) d\theta} &= \int \frac{q(\theta)}{\int q^*(\theta) d\theta} d\theta = \int \frac{q(\theta)}{f^*(\theta) \int q^*(\theta) d\theta} f^*(\theta) d\theta \\
&= \int \frac{q(\theta)}{\frac{q*(\theta)}{\int q^*(\theta) d\theta} \int q^*(\theta) d\theta} f^*(\theta) d\theta = \int \frac{q(\theta)}{q^*(\theta)} f^*(\theta) d\theta
\end{aligned} \tag{11}$$

Therefore, this can be rewritten on a form of which importance sampling can be performed over the H parameter samples from function $f^*(\theta)$ to approximate the distribution such that:

$$\int \frac{q(\theta)}{q^*(\theta)} f^*(\theta) d\theta \approx \frac{1}{H} \sum_{h=1}^{H} \frac{q(\theta_h)}{q^*(\theta_h)} \tag{12}$$

Using this approach, we can then insert this into our equation and use this to approximate the function of interest, $\mathbf{E}[g(\Theta)]$ through sampling posterior parameter draws:

$$\mathbf{E}[g(\Theta)] \approx \frac{1}{H} \sum_{h=1}^{H} p(z \mid Y_{1000} = y^*_{1000}, A, S, \theta_h) \frac{\frac{f(\theta_h \mid Y_{1000}=y^*_{1000},A,S)}{f^*(\theta_h \mid Y_{1000}=y_{1000},A,S)}}{\frac{1}{H}\sum_{k=1}^{H} \frac{f(\theta_k \mid Y_{1000}=y^*_{1000},A,S)}{f^*(\theta_k \mid Y_{1000}=y_{1000},A,S)}}, \quad \theta_1, \ldots, \theta_H \sim f^* \tag{13}$$

Conveniently, the distribution $f^*(\theta_h \mid Y_{1000} = y_{1000}, A, S)$ can be computed efficiently by fitting a linear model on the real dataset with the causal relations used for synthesis to get the H posterior parameter draws. These can then be used in the importance sampling step in Equation 13 to approximate $p(z \mid Y_{1000} = y^*_{1000}, A, S, \theta)$, efficiently.

When we fit a generalised linear model on the real dataset with the causal relations of synthesis, we then get the H parameter draws of Θ such that for each parameter draw we have $\theta_h = (\mu^{(h)}, \sigma_1^{(h)}, \alpha_1^{(h)}, \beta_1^{(h)}, \sigma_2^{(h)}, \alpha_2^{(h)}, \beta_2^{(h)}, \beta_3^{(h)}, \sigma_3^{(h)})$. We can then use these parameter draws to calculate the densities in our importance sampling in the following way:

$$\begin{aligned}
p(z \mid Y_{1000} = y^*_{1000}, A, S, \theta_h) &= \prod_{i=1}^{n} \left(\phi(z_{i1}, \mu^{(h)}, \sigma_1^{(h)}) \phi(z_{i2}, \alpha_1^{(h)} + \beta_1^{(h)} \cdot z_{i1}, \sigma_2^{(h)}) \phi(z_{i3}, \alpha_2^{(h)} + \beta_2^{(h)} \cdot z_{i1} + \beta_3^{(h)} \cdot z_{i2}, \sigma_3^{(h)}) \right) \\
f(\theta_h \mid Y_{1000} = y^*_{1000}, A, S) &= \phi(y^*_{1g}, \mu^{(h)}, \sigma_1^{(h)}) \phi(y^*_{2g}, \alpha_1^{(h)} + \beta_1^{(h)} \cdot y^*_{1g}, \sigma_2^{(h)}) \phi(y^*_{3g}, \alpha_2^{(h)} + \beta_2^{(h)} \cdot y^*_{1g} + \beta_3^{(h)} \cdot y^*_{2g}, \sigma_3^{(h)}) \\
f^*(\theta_h \mid Y_{1000} = y_{1000}, A, S) &= \phi(y_{i1}, \mu^{(h)}, \sigma_1^{(h)}) \phi(y_{i2}, \alpha_1^{(h)} + \beta_1^{(h)} \cdot y_{i1}, \sigma_2^{(h)}) \phi(y_{i3}, \alpha_2^{(h)} + \beta_2^{(h)} \cdot y_{i1} + \beta_3^{(h)} \cdot y_{i2}, \sigma_3^{(h)})
\end{aligned} \tag{14}$$

Here, $\phi(\cdot)$ is a probability density function (pdf) of a normal distribution, and $y^*_{1g}, y^*_{2g}, y^*_{3g}$ are the guesses being evaluated. These guesses are from a collection of guesses $\{y^*_{11}, \ldots, y^*_{1_{G1}}, y^*_{21}, \ldots, y^*_{2_{G2}}, y^*_{31}, \ldots, y^*_{3_{G3}}\}$, where G_1, G_2 and G_3 are the number of guesses for the real value of each variable, such that there are $G_1 \times G_2 \times G_3$ number of guesses. The guesses are constructed such that the true confidential value triplet (y_{i1}, y_{i2}, y_{i3}) is in the collection of guesses. Furthermore, since we use continuous variables, a neighbourhood interval of $[y_{ij} \times 0.9, y_{ij} \times 1.1]$ (i.e. within a 20% radius of y_{ij}) from which G_j equally spaced guesses are selected. Using these formulas, we can thereby estimate $g(\theta \mid Y_{1000} = y^*_{1000}, A, S)$, and as mentioned previously, $g(\theta \mid Y_{1000} = y^*_{1000}, A, S) = f(z \mid Y_{1000} = y^*_{1000}, A, S)$. This means that we can estimate the likelihood ($p(z \mid Y_{1000} = y^*_{1000}, A, S)$), which given that we have a uniform prior lets us estimate the posterior probability $p(Y_{1000} = y^*_{1000} \mid z, A, S)$.

4.2 PrivBayes for Generating Continuous Synthetic Data

For a method M to be DP, enough noise to satisfy Definition 1 is added when generating the synthetic dataset. One method that theoretically is ε-DP is PrivBayes [3]PrivBayes theoretically provides a ε-DP method by injecting noise into the construction and distributions of a Bayesian network, which they then can sample from without influencing the property of DP. However, Stadler et. al. [7] has shown that this is not the case for the actual implementation.

4.2.1 Noisy Network Construction. First, PrivBayes discretises the domain of continuous attributes by constructing a fixed number b of equi-width bins of the domain space of the variable. These bins are binary encoded such that if $b = 8$, the bins are represented by 3 (i.e. $log_2 8$) binaries such that we have the bins $\{(000), (001), \ldots (111)\}$. Continuous

attributes are then discretised by assigning them to these binary encoded bins according to what domain they are categorised as. They proceed by constructing a k-degree Bayesian network $\mathcal{B}$ over the h number of attributes in y, using an ε_1-DP method. To construct $\mathcal{B}$ in a differentially private manner, they start by randomly selecting an attribute X_1 from the set of attributes Att and use that as a root node such that $(X_1, \varnothing) \in \mathcal{B}$ and $X_1 \in V$, where V is the set of vertices in $\mathcal{B}$. A set containing all possible combinations of attributes and parent pairs (Ψ) is then made, such that:

$$\Psi = \left\{(X_i, \Pi) \mid X_i \in Att \setminus V, \Pi \in \binom{V}{k}\right\}, \tag{15}$$

where $\binom{V}{k}$ denotes the set of all subsets of V with size $min(k, |V|)$. Each attribute parent (AP) pair ($(X_i, \Pi) \in \Psi$) is then scored using a function $F(X, \Pi)$, which intends to approximate the mutual information between X and Π efficiently, such that

$$F(X, \Pi) = -\frac{1}{2} \min_{P^\diamond \in \mathcal{P}^\diamond} \left\| P(X, \Pi) - P^\diamond(X, \Pi) \right\|_1, \tag{16}$$

where $\mathcal{P}^\diamond$ is the set of all maximum joint distributions for X and Π, $P(X, \Pi)$ is the joint distribution of X and Π, and $\|\cdot - \cdot\|_1$ denotes the L_1 distance between the two element.

After that AP pairs (X_i, Π_i) are then sampled from Ψ, such that $\mathcal{B} = \mathcal{B} \cup \{(X_i, \Pi_i)\}$ and $V = V \cup \{X_i\}$ until $\Psi = \varnothing$ with a sampling probability of any pair (X, Π) being proportional to $exp(\frac{F(X,\Pi)}{2\Delta})$, where Δ is a scaling factor responsible for the construction being ε_1-differentially private. Therefore, we have that $\Delta = (d-1)\frac{S(F)}{\varepsilon_1}$, where $d = |A|$ and $S(F) = \frac{1}{n}$ meaning that the sensitivity of function F is $\frac{1}{n}$, with n being the number of tuples in D. After sampling Ψ, we are then left with a noisy construction of a Bayesian network $\mathcal{B}$.

4.2.2 Noisy Conditional Distribution. Second, PrivBayes uses a ε_2-DP algorithm to generate a set of conditional distributions of Y, such that for each AP pair (X_i, Π_i), the conditional distribution $P(X_i|\Pi_i)$ has a noisy version of it $P^*(X_i|\Pi_i)$.

To generate the noisy conditional distributions, we start by materialising the joint distribution ($P(X_i, \Pi_i)$). Laplace noise is then added to the distribution of all attributes in Att to get a noisy joint distribution of each attribute $P^*(X_i, \Pi_i)$ such that:

$$P^*(X_i, \Pi_i) = P(X_i, \Pi_i) + Lap(\frac{2(d-k)}{n\varepsilon_2}) \tag{17}$$

When generating the noisy joint distribution for each attribute, negative values are set to 0 and all values are normalised to maintain a total probability mass of 1. The $(d-k)$ noisy conditional distributions can then be constructed by deriving $P^*(X_i \mid \Pi_i)\ \forall i \in [1 \dots h]$ from $P^*(X, \Pi)$.

4.2.3 Synthetic Data Sampling. Third, PrivBayes uses the Bayesian network $\mathcal{B}$ and the p number of noisy conditional distributions to derive an approximate distribution from which they sample to generate z.

For continuous data, we have to convert the data that was discretised in the first step back to continuous data. This is done by sampling the attribute values from uniform distributions given the probabilities derived from $P^*(X_i \mid \Pi_i)$, such that for an attribute:

$$z_{ij} \sim \mathcal{U}(bin_a, bin_b), \quad where \quad a \in [1 \dots h-1], \quad b \in [2 \dots h] \tag{18}$$

Here, $bin_1, bin_2, \dots bin_h$ denotes the sample space where $bin_1 = min(\{y_{ij} \mid i \in [1 \dots n]\})$, $bin_h = max(\{y_{ij} \mid i \in [1 \dots n]\})$, where the probability of sampling from a given bin is derived from $P^*(X_i \mid \Pi_i)$. Furthermore, $\mathcal{U}(bin_a, bin_b)$ denotes a uniform distribution such that $bin_a \leq z_{ij} \leq bin_b$ with equal probability over the sample space.

5 ADR OF DP SYNTHETIC DATA

In this Section, we propose a solution to estimating the ADR of PrivBayes. To measure the risk, we propose a method for obtaining θ from the ε-DP method, PrivBayes.

When we want to estimate the ADR of PrivBayes, the problem of applicability, as mentioned in Section 3.5, becomes apparent. This is due to the Linear Regression Synthesizer in Section 3.4 including the parameters (θ) used for synthesis in its output, while PrivBayes does not. This is problematic, as the ADR measurement requires sampling θ parameters from the $f^*(\cdot)$ function to perform importance sampling. However, while PrivBayes does not directly provide θ, it does construct the noisy Bayesian network ($\mathcal{B}$) from which the synthetic data is sampled. Recall that the $f^*(\cdot)$ actually is a function that captures the probability distribution $p(\theta \mid Y_i = y_i, S)$. Here, S is knowledge about the synthesis process, which can be derived from PrivBayes through the noisy AP pairs, where for each attribute we have AP pairs (Π) such that $(X_i, \Pi_i) \forall\, i \in [1 \dots j]$. The AP pairs directly describe the conditional distributions from which the synthetic dataset is sampled. Therefore, we extract these as outputs from Privbayes, as shown in Figure 4.

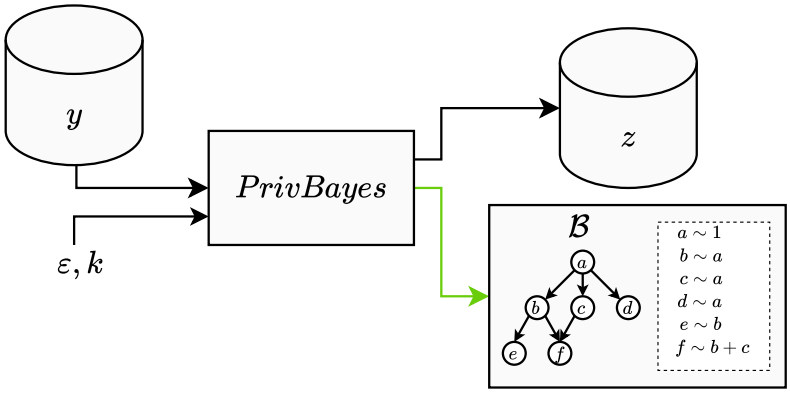

Fig. 4. Figure showcasing the inputs (y, ε, k) to PrivBayes, as well as the modification in output (indicated by the green arrow) from PrivBayes such that it both outputs the synthetic dataset (z) and causal relations in $\mathcal{B}$.

Given that we have the real dataset and the causal relations used to sample the synthetic dataset, we can use a generalised linear model to estimate the parameters (θ), and thereby calculate $p(\theta \mid Y_i = y_i, A, S)$. This can be done, as fitting a generalised linear model is a process such that for each attribute, we approximate $f^*(\theta \mid Y_i = y_i, A, S)$ by fitting a generalised linear model on the real dataset using the noisy conditionals of the Bayesian network from which we sample synthetic data. Thereby, by fitting a generalised linear model on the real dataset with the causal relations from PrivBayes, we can sample parameter draws from $f^*(\theta \mid Y_i = y_i, A, S)$.

Figure 5 showcases how we measure ADR of synthetic data from PrivBayes. The causal relations from PrivBayes, as well as the real dataset, are used in fitting a generalised linear model. To measure the ADR, the real dataset y, $\mathcal{B}$ and the synthetic dataset z from PrivBayes, as well as the parameters θ from the generalised linear model fitting, are used as input to the ADR measurement. The ADR is then calculated as elaborated in Section 4.1 such that we get the posterior probability $p(Y_i = y^* \mid z, A, S)$. This probability is assessed through multiple outputs denoted as risks in Figure 5. As can be seen from the Figure, the model does not discriminate whether the real dataset contains non-continuous

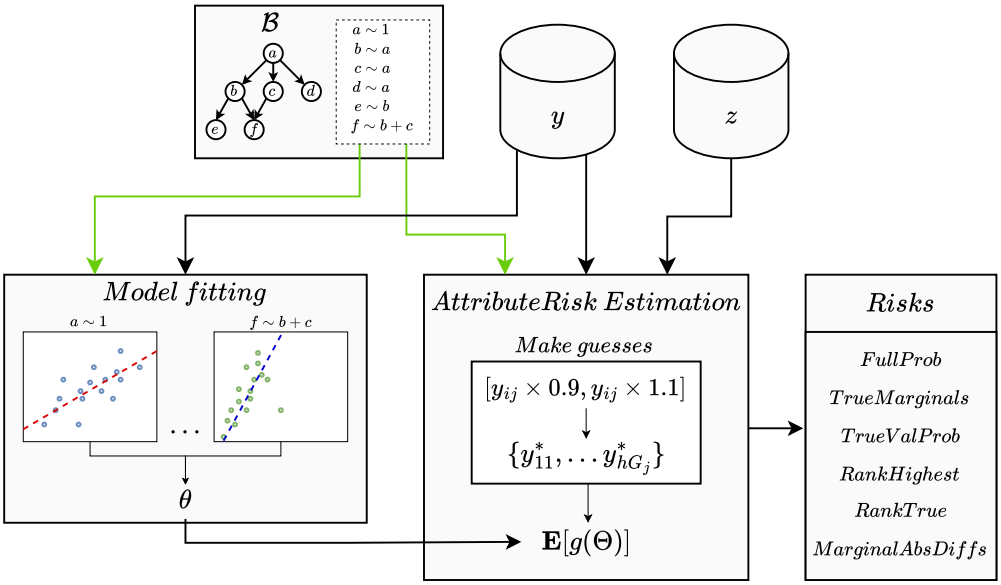

Fig. 5. Figure describing the process of measuring the ADR when incorporating the outputs from PrivBayes. Here, the modification to measuring the ADR is shown by the green arrows.

data, as the data can still be assessed. The assessment is, however, suboptimal, as the generalised linear model assumes a normal prior distribution, which may result in the parameters θ not being able to capture the distribution of the non-continuous attribute. This bad estimation of θ would then lead to a bad estimation of the true value of a synthetic attribute when importance sampling is performed.

A list of the different risk assessments can be found in Table 1. Using these, we can assert $p(Y_i = y^* \mid z, A, S)$ from different angles. Here, FullProb lets us observe the posterior probability of each of our guesses, while TrueMarginals and TrueValProb directly gives us the posterior probability of $p(Y_i = y^* \mid z, A, S)$. RankHighest and RankTrue provides a different angle to the assertion, as the guesses are ranked according to their posterior probability with the guess having the highest probability being rank 1. Using the rank, we can e.g. assert whether some records have a indication high ADR for a given attribute. Another way to assert how the probability of guessing correctly is through MarginalAbsDiffs which provides the distance from our posterior guess to the real value.

Table 1. A Table of the outputs obtained from ADR as well as a description thereof.

Output	**Description**
FullProb	*A list of posterior probabilities for each guess.*
TrueMarginals	*The posterior probability of the correct guess.*
TrueValProb	*The true value and the posterior probability of guessing that.*
RankHighest	*The guess with the highest posterior probability (The highest ranking guess).*
RankTrue	*The rank of the posterior probability of the guess being correct.*
MarginalAbsDiffs	*The absolute difference between the guessed value and the true value.*

6 EXPERIMENTS

In this Section, we present the experimental setup which includes a description of the datasets used for the experiments and details about how we use PrivBayes and measure ADR on the synthetic dataset. After this, we provide information on the individual experiments. For the following sections, we use the terms *prior* and *posterior* as shorthand for the prior probability of guessing correctly and the posterior probability of guessing correctly, respectively.

6.1 Datasets

For the experiments, we use three datasets. two datasets that contain continuous medical data about individuals and the dataset used in Hornby & Hu [11].

Dataset	#Tuples	#Attributes	Attribute types	Continuous attributes
CEData [11]	2000	5	binary, categorical and continuous	$\{LogIncome, LogExpenditure\}$
OL[1]	1000	16	binary, categorical and continuous	$\{Age, Height, Weight, NCP, CAEC, CH2O, FAF\}$
DPHP [2]	1000	24	binary, categorical and continuous	$\{AlcoholLevel, HeartRate, BloodOxygenLevel, BodyTemperature, Weight, MRI\_Delay, Age\}$

Table 2. The datasets used in the experiments after removing garbage attributes. Here, **#Tuples** and **#Attributes** are the number of tuples and attributes used respectively, while **Attribute types** are the types of attributes in the dataset, and **Continuous attributes** are the continuous attributes in the dataset.

Here, CEData is a collection of consumer expenditure data published by the U.S. Bureau of Labor Statistics. Here CEData has been cut from 5126 tuples to 2000 using random sampling due to computability of ADR. Furthermore, the attributes $\{TotalIncomeLastYear, TotalExpLastQ\}$ have been removed, as $\{LogIncome, LogExpenditure\}$ are the Log of these, meaning they would have a one-to-one correlation, which would induce problems when generating the noisy Bayesian network in PrivBayes as well as the measurement of ADR.

The OL (Obesity Levels) dataset is an estimation of obesity levels for individuals from Mexico, Peru and Colombia, based on their eating habits and physical condition. The dataset has been cut from 2111 tuples to 1000 using random sampling due to computability of ADR. Here, the attribute $\{NObeyesdad\}$ has been removed, as this attribute is merely used as a target label for classification purposes.

DPHP (Dementia Patient Health Prescriptions) is a dataset of risk factors, that may contribute to the onset and progression of dementia in patients. In this dataset, the attributes $\{Prescription, Dosageinmg\}$ have been removed, as these had missing attribute values, due to ADR's inability to handle $NULL$ values.

6.2 Experimental Setup

For measuring ADR, we use PrivBayes to generate a synthetic variant of the real dataset (both continuous and non-continuous variables), as elaborated in Section 4.2. From this, we obtain $\mathcal{B}$ through the noisy causal relations of PrivBayes as well as z, where $|z| = |y|$.

Here the default network degree of the noisy Bayesian network $k = 2$, and the number of equally spaced values used for guessing the real value of an attribute in ADR is $G_j = 41$, and the prior $1/41$ for all guesses.

When the synthetic dataset and noisy causal relation have been obtained, we fit a linear model for each attribute (both continuous and non-continuous) using the noisy causal relations to estimate the posterior parameter draws θ.

[1] https://www.kaggle.com/datasets/fatemehmehrparvar/obesity-levels
[2] https://www.kaggle.com/datasets/kaggler2412/dementia-patient-health-and-prescriptions-dataset/data

These as well as z, y and the noisy causal relations are then used to estimate the ADR through the outputs presented in Table 1 using the approach elaborated in Section 4.1.

For each experiment, we have multiple experimental conditions, such as different ε-value in Experiment 1. To account for potential variance that might occur, both for PrivBayes and the ADR estimation, each experimental condition is run three times. However, due to an oversight, this did not happen as elaborated in Section 9.

6.3 Experiment 1: ε and ADR correlation

In this experiment, we investigate the possible correlation, between ε and ADR. We hypothesise there is a positive correlation between the ε-value and the risk for attribute disclosure, such that a low ε-value will correspond to a low ADR. To test this, we generate three synthetic datasets for each $\varepsilon \in \{0, 0.05, 0.1, 0.2, 0.5, 1, 5, 10, 100, \infty\}$ using PrivBayes, where 0 represents our prior probability of guessing correctly, which is what we would expect our posterior probability being when adding infinite noise, and ∞ represents PrivBayes where noise has been removed from the synthesis process. With these values, we measure attribute disclosure probabilities between these ε-values, for all three datasets mentioned in 2. Using these measurements, we analyse the correlation between ε and ADR by averaging the posterior probability of guessing correct for each run and ε. Here, the average ADR of continuous attributes for each ε-value of a given real dataset is thereby:

$$avgADR = \frac{1}{r}\sum_{r=1}^{R}\frac{1}{n}\sum_{i=1}^{n} p(Y_i = y^* \mid z, A, S), \tag{19}$$

where R is the number of runs for each ε-value and n is the number of tuples in the real dataset (y). From the outputs of ADR estimation in Table 1, the probability $p(Y_i = y^* \mid z, A, S)$ is given in the $TrueMarginals$ variable. To analyse the correlation between ε and $avgADR$, we will produce plots with ε as the x-axis and $avgADR$ as the y-axis for each real dataset, and from this, the possible correlation can be analysed.

Furthermore, another plot will be produced to estimate the correlation between ADR and ε from another point of view. Here, we analyse the density of the posterior probability of guessing correct from the output variable $TrueValProb$ in Table 1 for some continuous attributes, and compare this to both the prior probability of guessing correct assuming a naive prior and the different ε-values.

6.4 Experiment 2: Extreme outlier injection

Here, we investigate the effects of injecting outliers into the real dataset. More specifically by injecting we mean, replacing the first individual with an outlier. Here, we have two experimental conditions:

The first experimental condition, is a fictive outlier injected into the real dataset, which has all attribute values being the max values of the real dataset, meaning that for the individual y_{IN} we have that:

$$y_{IN} = \left\{ \max_{i \in [1 \ldots n]} (y_{ij}) \; \forall \; j \in [1 \ldots h] \right\} \tag{20}$$

We then produce a synthetic dataset for each $\varepsilon \in \{0, 0.05, 0.1, 0.2, 0.5, 1, 5, 10, 100, \infty\}$ using PrivBayes, where 0 represents our prior probability of guessing correctly, which is what we would expect our posterior probability being when adding infinite noise, and ∞ represents PrivBayes where noise has been removed from the synthesis process. Following that, we estimate the ADR of the synthetic datasets, and investigate whether this individual (y_{IN}) has a significant ADR.

Second, we have an experimental condition where we inject a fictive outlier with attribute values outside the real dataset, where all continuous attribute values being the max values of the real dataset multiplied by 1.1, while also

introducing a new value for categorical attributes, where $y_{ij} = y_{ij} + 1$, meaning that for the continuous attributes of individual y_{OUT}, we have that:

$$y_{OUT} = \left\{ \max_{i \in [1 \dots n]} (y_{ij} \cdot 1.1) \; \forall \; j \in [1 \dots cont] \right\} \cup \left\{ \max_{l \in [1 \dots n]} (y_{il} + 1) \; \forall \; l \in [1 \dots cat] \right\}, \tag{21}$$

where *cont* and *cat* are the number of continuous and categorical attributes, respectively. We then produce a synthetic dataset for each $\varepsilon \in \{0, 0.05, 0.1, 0.2, 0.5, 1, 5, 10, 100\}$ using PrivBayes. Following that, we estimate the ADR of the synthetic datasets, and investigate whether this individual (y_{IN}) has a significant ADR.

The significance of the ADR for fictive individuals y_{IN} and y_{OUT} will be estimated through the output $TrueMarginals$ as well as the $RankTrue$ outputs in Table 1. This will be done by looking at these outputs for the individuals y_{IN} and y_{OUT}.

We hypothesise that due to the properties of DP, the individual y_{IN} will have a low ADR for low ε-values, while the ADR might increase as the ε-value increases, as setting the ε-value to high will mean that PrivBayes will no longer satisfy the properties of DP. In the case of individual y_{OUT}, we hypothesise that it will mostly likely be the same case as for individual y_{IN}. There is however a possibility that there will be a higher ADR for an individual y_{OUT}. This is due to the problems in the implementation of PrivBayes, mentioned by Stadler et al. [7]. Here, they discover that the implementation of PrivBayes is not ε-DP due to them using information from the real dataset after introducing noise. This is also apparent from the description of how PrivBayes generates continuous synthetic data in Section 4.2. Here, it is apparent that when sampling the synthetic dataset, they sample from uniform distributions over bins, where the bins bin_1 and bin_h have access to information about the maximum and minimum attribute values. Therefore, by introducing the fictive outlier y_{OUT} in the real dataset, we expand the sample space of the synthetic data sampling, which may lead to a higher ADR.

7 RESULTS

In this section, we present our findings from running the experiments from Section 6. Here, we present an analysis of the datasets generated using PrivBayes, followed by an analysis of the results of these experiments. The results presented here will be further discussed in Section 8.

7.1 Synthetic Datasets From PrivBayes

PrivBayes, as discussed in Section 4.2, injects noise into both the construction and marginals of the Bayesian network, from which the synthetic data is sampled. Furthermore, they discretise continuous variables using bins, and sample from a uniform distribution for each bin.

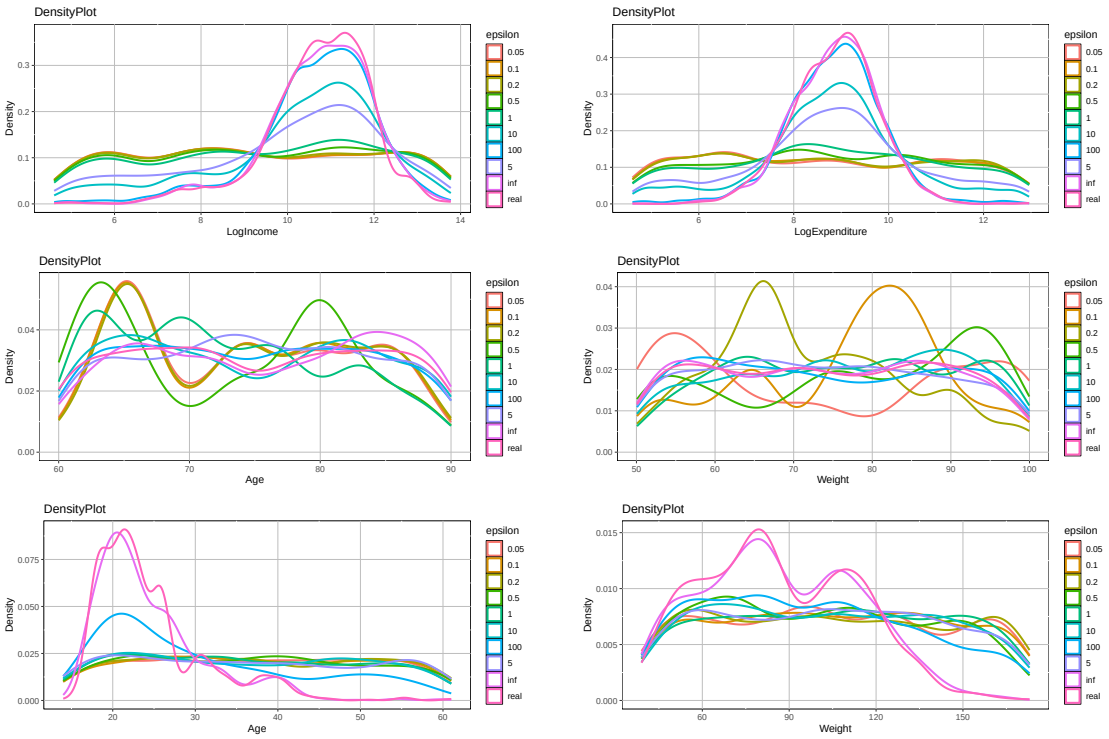

Fig. 6. Figure showcasing the distribution of six continuous attributes for the real and synthetic datasets given different ε. Here, the top two plots are for attributes $\{LogIncome, LogExpenditure\}$ from CEData, the next two are $\{Age, Weight\}$ from DPHP and the last two attributes are $\{Age, Weight\}$ from the OL dataset.

The plots in Figure 6 show that, in general, PrivBayes is able to produce synthetic datasets where the distribution of attributes deviates more than the distribution from the real dataset when $\varepsilon \to 0^+$. When attributes have distributions similar to a normal distribution, we observe that the distribution becomes more uniform for $\varepsilon \to 0^+$, which is expected when noise is injected. For attributes with a more uniform distribution, we see that the distribution fluctuates more and therefore deviate from the distribution of the real dataset given that $\varepsilon \to 0^+$. Given these observations, we see that using PrivBayes to work as expected.

7.2 Experiment 1: ε and ADR correlation

For Experiment 1, we hypothesised that there is a correlation between the average probability of guessing correctly (*avgADR*) and ε.

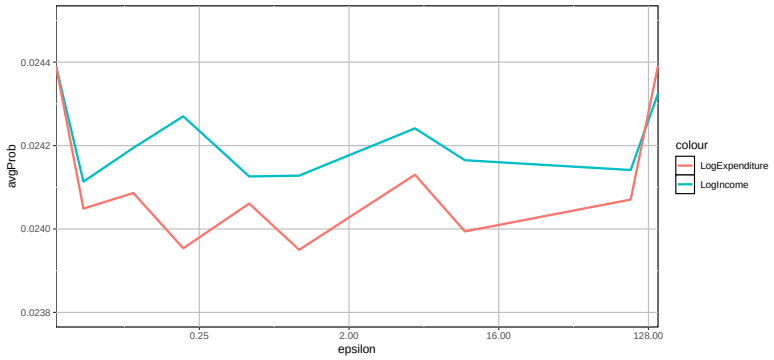

Fig. 7. Average probability (avgProb = *avgADR*) of guessing continuous variables correctly, for real and synthetic data using various ε for the CEData dataset. Here the first data point ($\varepsilon = 0$) in the graph is our prior ($1/41$), and the last point is $\varepsilon = \infty$ meaning that, there is no noise from PrivBayes injected via. ε.

However, Figure 7 shows a general flat trend in the *avgADR* as $\varepsilon = 0^+ \rightarrow \infty$ with some variance between data points. Therefore, it is not apparent whether there is a correlation between the average probability of guessing correctly (*avgADR*) and the ε-value.

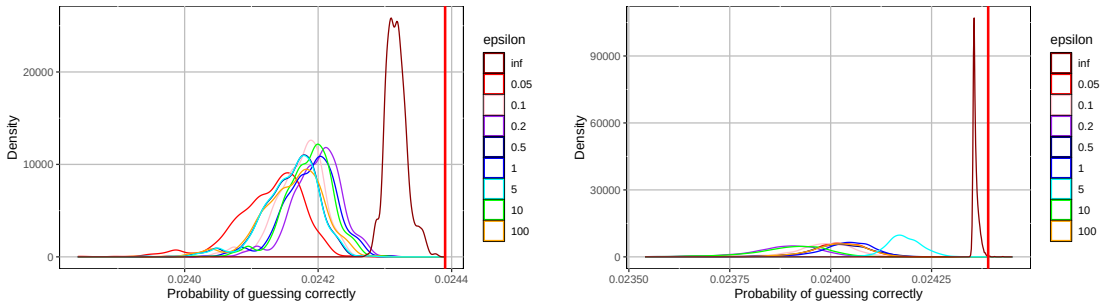

Fig. 8. Density of the probability of guessing continuous variables correctly for attributes *LogIncome* (left) and *LogExpenditure* (right) for the CEData dataset, which have been synthesised using different ε. Here the red vertical line ($\varepsilon = 0$) is our prior ($\frac{1}{41}$), and $\varepsilon = inf$ ($\varepsilon = \infty$) meaning that, there is no noise from PrivBayes injected via. ε.

Similarly, Figure 8 shows a lack of correlation with the density of the probability of guessing correctly for individual attributes in the CEData dataset. Here, we observe that using PrivBayes with no injected noise ($\varepsilon = \infty$), the ADR performs worse than the prior ($\frac{1}{41}$). Comparing this performance to the results from Hornby & Hu, who obtained a greater probability of guessing correctly, using a synthetic dataset from their linear regression synthesiser. This gives us some hints as to the reasoning behind ADR's performance, which will be discussed in Section 8.

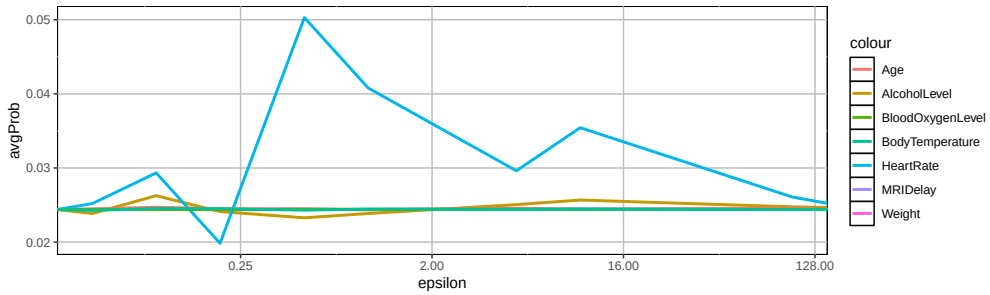

Fig. 9. Average probability (avgProb = *avgADR*) of guessing continuous variables correctly, for real and synthetic data using various ε for the DPHP dataset. Here the first data point ($\varepsilon = 0$) in the graph is our prior ($1/41$), and the last point is $\varepsilon = \infty$ meaning that, there is no noise from PrivBayes injected via. ε.

For the *avgADR* of the DPHP dataset, Figure 9 shows a generally flat trend with some variability between data points except for attributes *AlcoholLevel* and *HeartRate* where high seemingly random variance between points. Here, for the *HeartRate* attribute, it is debatable whether the attribute is continuous, because there only exists 41 unique values in the real dataset, which affects the estimation of $g(\Theta)$, as elaborated in Section 8.

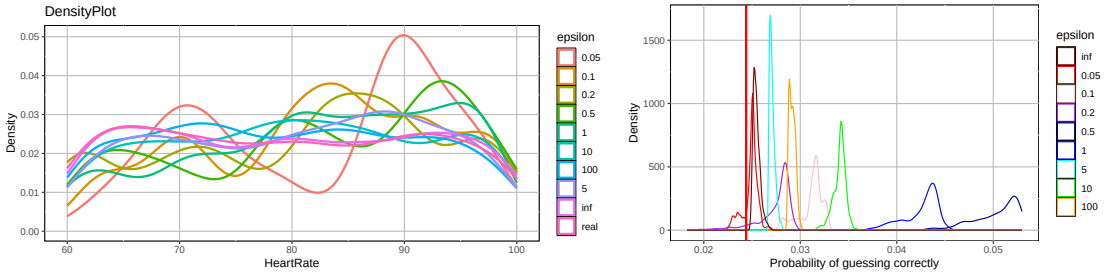

Fig. 10. Distribution of continuous attribute *HeartRate* from the DPHP dataset and for each ε used produce the synthetic version (left). As well as the density of the probability of correctly guessing a continuous attribute *HeartRate* of the DPHP dataset, which have been synthesised using different ε values (right). Here the red vertical line ($\varepsilon = 0$) in the graph is our prior between the 41 guesses ($\frac{1}{41}$), and $\varepsilon = inf$ ($\varepsilon = \infty$) meaning that, there is no noise from PrivBayes injected via. ε.

From the density of the posterior for the attribute *HeartRate* of the DPHP dataset, as shown in Figure 10, we see generally better performance, although there is no observable correlation between ε and *avgADR*. Despite this, the densities of our posterior are still significantly higher for *HeartRate* when $\varepsilon \in \{0.5, 1\}$. This may be due to run bias, as elaborated in Section 9, which means that this may be due to a good seed for this exact run.

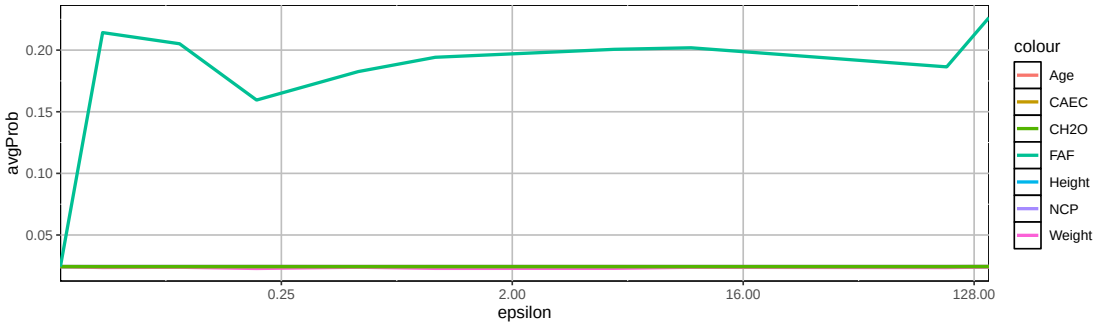

Fig. 11. Average probability (avgProb = *avgADR*) of guessing continuous variables correctly, for real and synthetic data using various ε for the OL dataset. Here the first data point ($\varepsilon = 0$) in the graph is our prior ($1/41$), and the last point is $\varepsilon = \infty$ meaning that, there is no noise from PrivBayes injected via. ε.

For the *avgADR* of the OL dataset, as shown in Figure 11 we again see a flat trend, with minor variations between data points, except for the attribute *FAF*, where we see a high *avgADR* for all ε-values. We suspect this may be due to the same reasons as for the other datasets.

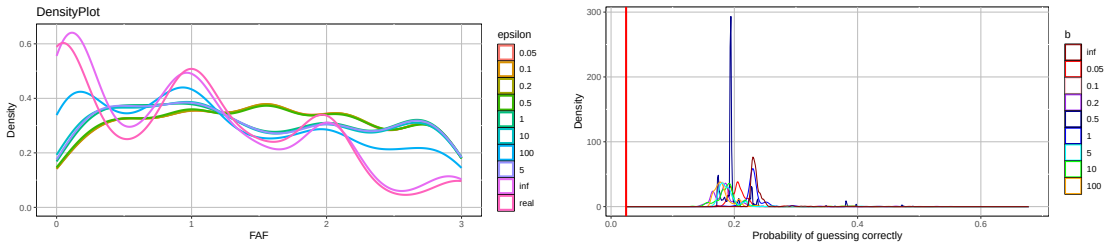

Fig. 12. Distribution of continuous attribute *FAF* for the real OL dataset and synthetic variations using various ε (left). As well as the density of the probability of correctly guessing a continuous attribute *FAF* of the OL dataset, which have been synthesised using different ε values (right). Here the red vertical line ($\varepsilon = 0$) in the graph is our prior between the 41 guesses ($\frac{1}{41}$), and $\varepsilon = inf$ ($\varepsilon = \infty$) meaning that, there is no noise from PrivBayes injected via. ε.

Furthermore, for the *FAF* attribute, we observe that the distribution is mostly concentrated on four exact values, i.e. 0, 1, 2, 3, with some other in-between values, as shown in Figure 12. Thereby, we suspect that *FAF* has the same issues as the *HeartRate* attribute i DPHP.

Summarizing Experiment 1, we see no correlation between the value of ε and the probability of guessing correctly, suggesting that our hypothesis should be rejected. A possible reason could be computational, as small datasets and parameter values lead to faster but less accurate results. We further discuss this in Section 9. We also observe that, for most attributes, the posterior does not differ in a statistically meaningful way compared to the prior. This minor difference in probabilities for the prior and posterior, correlate with Hornby & Hu [11], who also demonstrated results of similar magnitude.

7.3 Experiment 2: Extreme outlier injection

For this experiment, we hypothesised that, like in Experiment 1, there is a correlation between epsilon and ADR for individuals, but that the ADR for outliers y_{IN}, y_{OUT} would be significantly higher than the normal individual. Here, we suspect that the ADR for an individual y_{OUT} might be higher than y_{IN}, due to how PrivBayes handles sampling, as elaborated in Section 6.4.

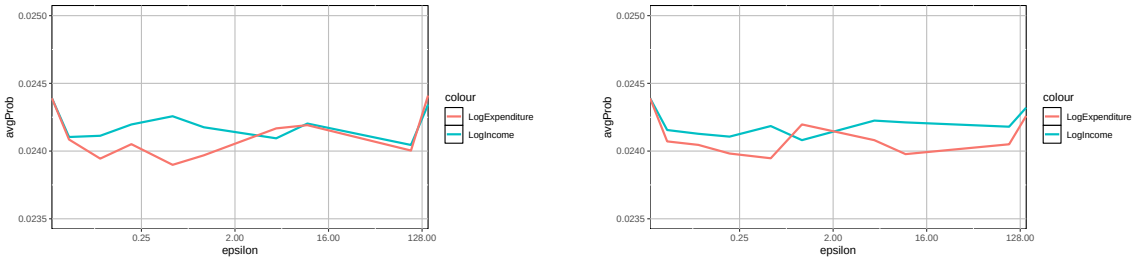

Fig. 13. average probability of guessing continuous variables correctly, for real and synthetic data using various ε for the CEData dataset the individual y_{IN} (left) and y_{OUT} (right). Here the first data point ($\varepsilon = 0$) in the graph is our prior (1/41), and the last point is $\varepsilon = \infty$ meaning that, there is no noise from PrivBayes injected via. the ε-value.

From Figure 13, we can see that there is no significant difference in the *avgADR* compared to the *avgADR* for the CEData dataset without y_{IN} and y_{OUT}. When observing the $TrueMarginals$ for continuous attribute $\{LogIncome, LogExpenditure\}$, we see no increase in the probability of guessing correctly for individuals y_{IN} and y_{OUT} when compared to the *avgADR* of the CEData dataset without the individuals. Even though this is the case, when estimating the ADR, we did it for all attributes.

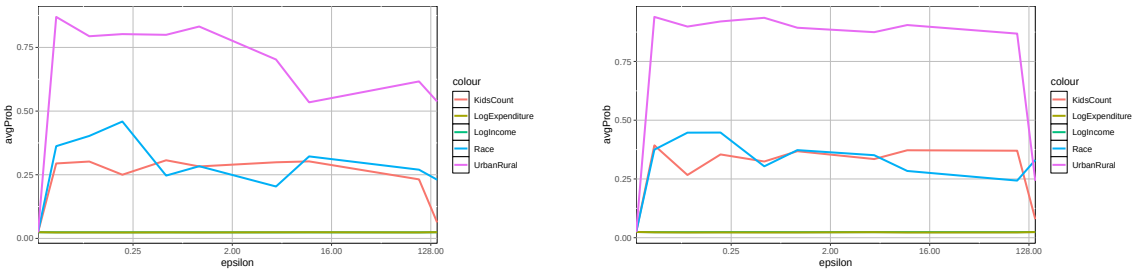

Fig. 14. The probability (avgProb = *avgADR*) of guessing variables of the specific individuals y_{IN} (left) and y_{OUT} (right) correctly for synthetic datasets using different ε for the CEData dataset with the individuals included. Here the first data point ($\varepsilon = 0$) in the graph is our prior (1/41), and the last point is $\varepsilon = \infty$ meaning that, there is no noise from PrivBayes injected via. ε.

As can be seen in Figure 14, the probability of guessing the correct values for continuous attributes of individuals y_{IN} and y_{OUT} is not influenced by the individuals being outliers or the ε-value used for synthesis. It is, however, not the case for guessing correctly for categorical attributes. Here, we are able to correctly guess the attribute values of y_{IN} and y_{OUT} in almost all instances, where our average probability of guessing correctly for the attribute $UrbanRural$ is nearly 100% for y_{OUT}.

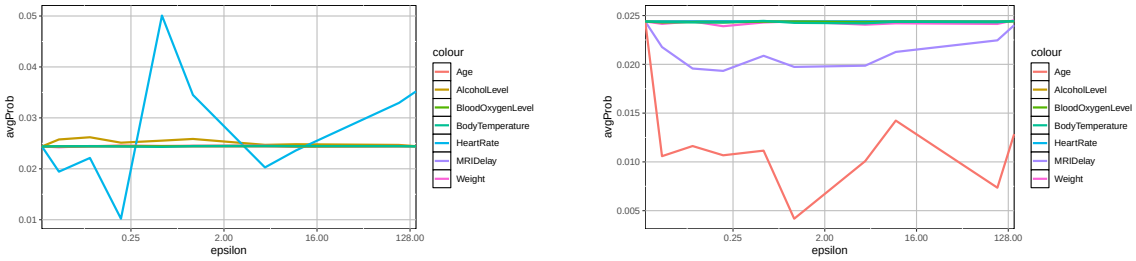

Fig. 15. Average probability (avgProb = *avgADR*) of guessing continuous variables correctly for the synthetic variations, using different ε for the DPHP dataset with y_{IN} (left) and y_{OUT} (right). Here the first data point ($\varepsilon = 0$) in the graph is our prior ($1/41$), and the last point is $\varepsilon = \infty$ meaning that, there is no noise from PrivBayes injected via. ε.

Figure 15 shows that there is a difference in the *avgADR* of the dataset with y_{OUT}, while there is an insignificant difference in *avgADR* of the dataset with y_{IN}, when compared to the *avgADR* for the DPHP dataset without y_{IN} and y_{OUT}. From this, we observe a decrease in *avgADR* for the dataset with y_{OUT}. This could be due to a bad seed, as elaborated previously. It could, however, also be due to the sampling used in PrivBayes, where we make the number of unique values for attributes larger by injecting the outlier y_{OUT}.

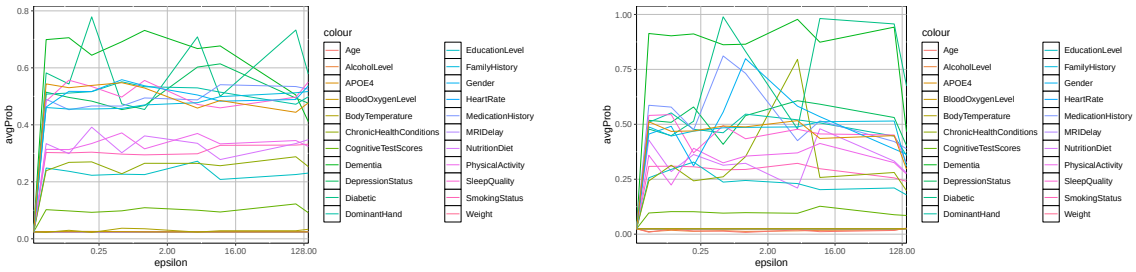

Fig. 16. The probability (avgProb = *avgADR*) of guessing variables of the specific individuals y_{IN} (left) and y_{OUT} (right) correctly for synthetic datasets that have been synthesised with different ε values of the DPHP dataset with the individuals in them. Here the first data point ($\varepsilon = 0$) in the graph is our prior ($1/41$), and the last point is $\varepsilon = \infty$ meaning that, there is no noise from PrivBayes injected via. the ε-value.

When looking at the *TrueMarginals* for the continuous attributes, we do not observe any increase in the probability of guessing correctly for individuals y_{IN} and y_{OUT} when compared to the *avgADR* of the DPHP dataset without the individuals. From Figure 16, we experience similar results as for the CEData dataset with outliers y_{IN} and y_{OUT} in it, as we see a high probability of guessing the categorical outlier attribute values, although there is no significant influence when $\varepsilon \longrightarrow 0^+$.

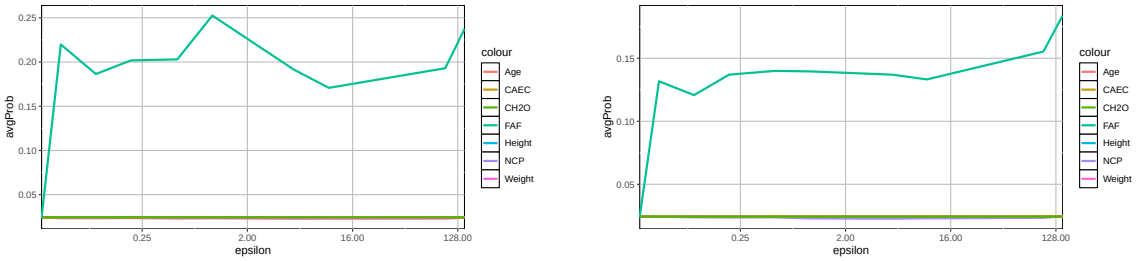

Fig. 17. Average probability (avgProb = *avgADR*) of guessing continuous variables correctly for the synthetic variations, using different ε for the OL dataset with y_{IN} (left) and y_{OUT} (right). Here the first data point ($\varepsilon = 0$) in the graph is our prior ($1/41$), and the last point is $\varepsilon = \infty$ meaning that, there is no noise from PrivBayes injected via. ε.

As can be seen in Figure 17, there is no significant difference in the *avgADR* compared to the *avgADR* for the OL dataset without y_{IN} and y_{OUT}. When inspecting the $TrueMarginals$ for continuous attributes, we also observe no increase in the probability of guessing correctly for individuals y_{IN} and y_{OUT} when compared to the *avgADR* of the OL dataset without the individuals.

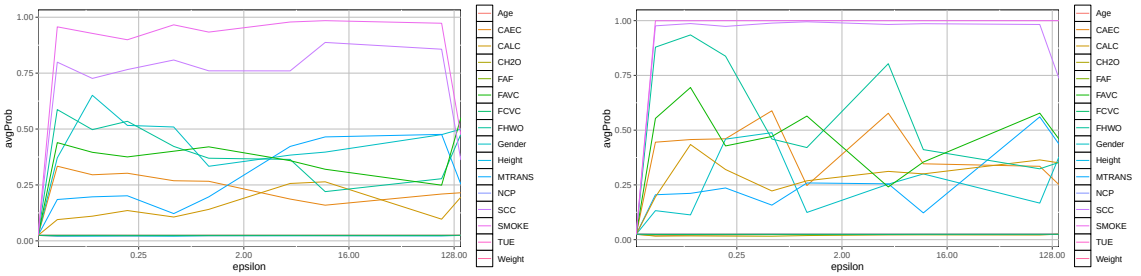

Fig. 18. The probability (avgProb = *avgADR*) of guessing variables of the specific individuals y_{IN} (left) and y_{OUT} (right) correctly for synthetic datasets using ε values for the OL dataset with the individuals in them. Here the first data point ($\varepsilon = 0$) in the graph is our prior ($1/41$), and the last point is $\varepsilon = \infty$ meaning that, there is no noise from PrivBayes injected via. the ε-value.

From Figure 18, we observe that the probability of guessing the correct values for continuous attributes of individuals y_{IN} and y_{OUT} is not significantly influenced by the individuals being outliers or by the ε used. It is, however, apparent that this is not the case for guessing correctly for categorical attributes. Here, we are able to correctly guess the attribute values of y_{IN} and y_{OUT} in almost all instances, where the average probability of guessing correctly for the categorical attributes $\{SMOKE, SCC\}$ is almost 100% for both y_{IN} and y_{OUT}.

Summarizing the results for Experiment 2, we see that injecting outliers y_{IN} and y_{OUT} did not provide any significant increase in the *avgADR* nor our probability of guessing continuous attribute values of y_{IN} and y_{OUT}. This was, however, not the case for categorical attributes for these individuals, as the *avgADR* of these attributes is able to accurately guess the correct attribute values for most attributes, while the ε-value used for synthesis did not significantly decrease the *avgADR* when $\varepsilon \longrightarrow 0^+$ as with the results for Experiment 1.

8 DISCUSSION

In terms of the estimation of ADR of synthetic datasets generated by PrivBayes, as elaborated in Section 4.2, PrivBayes samples a synthetic dataset from multiple bins of uniform distributions. This could have a great effect on the utility of the synthetic dataset, as the statistical properties of the distribution of attributes will lose nuance, when injecting unnecessary uniform noise through bin sampling. This means that when we estimate ADR, as elaborated in Section 4, it is more difficult to observe the influence of ε when using the synthetic datasets from PrivBayes. This is apparent when estimating $g(\Theta)$, as $p(z \mid Y_i = y_i^*, A, S, \theta)$ influences our estimation, as shown in Equation 14. This is due to the possible problem that when sampling the density $f^*(\theta \mid Y_i = y_i^*, A, S)$, the fact that we sample parameter draws using a generalised linear model fitted on the real data may mean that these parameter draws possibly do not contain the necessary information to estimate $g(\theta)$. Furthermore, Hornby & Hu produce their synthetic dataset directly from the model fitted on the real dataset. Meaning that they store information about the distribution of both the real and synthetic datasets in Θ. This will typically result in better estimates of the attributes in the real dataset, given the synthetic dataset through Θ. This we are unable to do, as we do not sample data directly from the fitted linear model, which could explain a lower posterior. Therefore, we encourage better estimation of Θ, where one could incorporate the noise injected into the SDG in the estimates. Here, one could look into what is done by Mehnaz et al. [13], where they try to estimate sensitive attributes by learning parameters of a model.

For both Experiment 1 and Experiment 2, we chose $\varepsilon \in \{0, 0.05, 0.1, 0.2, 0.5, 1, 5, 10, 100, \infty\}$, where $\varepsilon = 0$ was our prior, and $\varepsilon = \infty$ was when no noise was injected in PrivBayes. This aligns with how larger corporations seem to use ε-values within the same range for privacy preservation [20]. The scale of the data they protect, however, is numbers of magnitude larger than the datasets we tested. This means that two neighbouring datasets from our testing will vary significantly more in distribution than the datasets of the companies. Therefore, we suspect that even the lowest ε-value used in the experiments (0.05) might not inject enough noise to preserve privacy, and that, smaller ε-values are worth investigating.

The results that we obtained from running these experiments presented in the Section 7, showed us that coupling PrivBayes to Hornby & Hu, produced results that show a mostly insignificant change between guessing randomly, and the guessing performed in ADR. This aligns with results that Hornby & Hu provide in their paper. Reflecting back on Hornby & Hu, it becomes apparent that their work had many issues when estimating ADR for continuous attributes. First, they demonstrate limited testing by only documenting one execution of their package for one dataset, and therefore does not demonstrate that their results are consistent and repeatable. Second, is their low experimental performance in which their prior and posterior only differ by a statistically insignificant amount, which our results also demonstrate in Section 7. Last is their incomplete documentation, where one example was that there was no single file for executing the code, and that we had to piece together code snippets, to be able to run their code.

In Experiment 2, while we were unable to guess continuous attribute values, we observed that we were able to correctly guess most categorical attribute values of both y_{IN} and y_{OUT}. This is in accordance with our hypothesis for these individual. This observation tells us that the problems observed by Statler et al. [7] of a DP violation in the synthetic data sampling of PrivBayes may be correct. Therefore, a better sampling for PrivBayes should be researched.

Regarding the limitations mentioned in Section 9, the calculation of ADR is slow, which apprehended us from calculating ADR for larger datasets. The usage of larger datasets could help us provide estimates of the ADR in different scenarios, where the adversary would have more knowledge about A. Here, we could, e.g., investigate the influence of outlier injection on various dataset sizes. Furthermore, this also resulted in the inability to perform tests using a high

number of iteration of the ADR estimation, where we have used the same number as Hornby & Hu (40). Since we were unable to test this, we can not conclude with certainty that this will improve the results.

9 LIMITATIONS

In this section, we discuss the limitations and the impact they had on the results.

Due to an oversight on our part, the seed for PrivBayes was a fixed value meaning that the variance for synthetic data generation was not captured, meaning that there is almost no variation in *avgADR* between the three runs. This could explain why there was no observed correlation between ε and the chance of guessing correctly.

In Experiment 2, the outliers replaced an individual, instead of being appended, this makes the outlier datasets less comparable to the original real datasets, as you could be replacing an outlier with another outlier for one dataset, and an individual with the most common attribute values for another dataset. Were we to redo the experiment, the outlier would be appended to the dataset instead of replacing an individual.

Another limitation is the computation speed of Hornby & Hu, as it lacks GPU and/or multithreading support. R as a language is generally considered slow, making it a poor choice for a Bayesian privacy metric, as functions typically used within Bayesian statistics like Monte Carlo estimation are computationally expensive, which limits our ability to test on larger datasets.

10 CONCLUSION

In this paper, we presented a coupling between PrivBayes and Hornby & Hu. Hornby & Hu, seemed promising due to its ability to measure the risk of attribute disclosure, given the synthetic data, real data, auxiliary information an adversary might know and information about the synthetic data generation, with the latter two inputs being relatively novel in the domain of private synthetic data. However, despite this extra information, they only demonstrate minor changes to the chance of guessing continuous attribute values correctly in their testing. This is something we tested more extensively with PrivBayes as the synthesiser, and our results, like theirs, show little change in ADR for continuous attributes, which is unexpected considering the additional information available to the adversary. Furthermore, the results also demonstrated that the ADR for continuous attributes was not directly influenced by the amount of noise injected by PrivBayes. This could indicate that this additional information might be insignificant or distracting when using PrivBayes, making guessing correctly harder. Despite this, we believe that Bayesian modelling provides a promising estimation where we are able to adjust the knowledge available to the adversary, which can provide more accurate results in the vision of protecting individuals' sensitive attributes.

11 FUTURE WORKS

In our estimation of ADR, we assume that the prior probability $p(Y_{1000} = y^*_{1000} \mid A, S)$ is uniform such that $p(Y_{1000} = y^*_{1000} \mid A, S) = 1/n$ for all y in the support. This, however, means that our (as well Hornby & Hu's) prior lacks information about A and S. Such information might be useful in the estimation process, as it could make the attack stronger. The additional information could be modelled in various ways to simulate different scenarios. Therefore, an interesting area of exploration would be to estimate a prior that incorporates information about A and S into the estimation of ADR.

Even though our estimation of ADR focused on continuous attributes, we see that ADR performed well on categorical attributes, where for some cases it got a nearly 100% accurate guess of the real attribute values of the fictional outliers (e.g., the $UrbanRural$ attribute in CEData). Therefore, it seems that ADR is able to identify certain attribute values that should remain private under DP, We therefore encourage further research into the estimation of ADR for both continuous and categorical attributes that have been synthesised by a DP-SDG.

In terms of the computation speed of estimating ADR, we suspect that there is ample room for optimization. Therefore, investigating Hornby & Hu's speed, scalability and accuracy in the context of other attribute inference attacks, like those presented in related work, could provide insight into what knowledge is the most significant for an adversary when executing an attribute inference attack, which is useful for developing stronger privacy attacks.

Furthermore, due to ADR's general poor performance in speed as well as accuracy despite the access to auxiliary information and information about the synthesisation method, investigating these aspects in context of other attribute inference metrics, could give insight on how to strengthen attribute inference attacks and what information is the most significant.

For DP, it remains unknown whether the increase in privacy preservation imposed by the properties of DP is cancelled out by the loss of utility these impose. There are no guidelines for the choice of the ε-value, as estimating ε imposes many difficult challenges. We however believe that Bayesian analysis poses a promising step in the direction of estimating the ε-value, and we thereby encourage others to work towards this goal. Here, we also encourage an investigation of other DP-SDGs, as it was apparent that the synthetic data sampling violated DP by observing the min and max values, and that the utility of the synthetic data, was negatively affected by the sampling.

ACKNOWLEDGMENTS

We would like to express our gratitude to our supervisors, Daniele Dell'Aglio, Martin Bøgsted, and Jakob Bruhn Krøjgaard Skelmose for their guidance, insight, expertise, and support throughout the duration of this project. Their encouragement and patience helped us navigate through this thesis.

REFERENCES

[1] Liyang Xie, Kaixiang Lin, Shu Wang, Fei Wang, and Jiayu Zhou. Differentially private generative adversarial network, 2018.

[2] Jinsung Yoon, James Jordon, and Mihaela van der Schaar. PATE-GAN: Generating synthetic data with differential privacy guarantees. In *International Conference on Learning Representations*, 2019.

[3] Jun Zhang, Graham Cormode, Cecilia M. Procopiuc, Divesh Srivastava, and Xiaokui Xiao. Privbayes: Private data release via bayesian networks. *ACM Trans. Database Syst.*, 42(4), oct 2017.

[4] Karan Bhanot, Miao Qi, John Erickson, Isabelle Guyon, and Kristin Bennett. The problem of fairness in synthetic healthcare data. *Entropy*, 23:1165, 09 2021.

[5] Datacebo. Synthetic data vaultdifferentially private generative adversarial network, 2024.

[6] Chao Yan, Yao Yan, Zhiyu Wan, Ziqi Zhang, Larsson Omberg, Justin Guinney, Sean D. Mooney, and Bradley A. Malin. A multifaceted benchmarking of synthetic electronic health record generation models. *Nature Communications*, 13(1), dec 2022.

[7] Theresa Stadler, Bristena Oprisanu, and Carmela Troncoso. Synthetic data - A privacy mirage. *CoRR*, abs/2011.07018, 2020.

[8] Cynthia Dwork and Moni Naor. On the difficulties of disclosure prevention in statistical databases or the case for differential privacy. *Journal of Privacy and Confidentiality*, 2(1), Sep. 2010.

[9] Jingchen Hu. Bayesian estimation of attribute and identification disclosure risks in synthetic data, 2021.

[10] Jerome P. Reiter, Quanli Wang, and Biyuan Zhang. Bayesian estimation of disclosure risks for multiply imputed, synthetic data. *Journal of Privacy and Confidentiality*, 6(1), Jun. 2014.

[11] Ryan Hornby and Jingchen Hu. Bayesian estimation of attribute disclosure risks in synthetic data with the `AttributeRiskCalculation` r package, 2021.

[12] Kevin Zhang, Neha Patki, and Kalyan Veeramachaneni. Sequential Models in the Synthetic Data Vault, July 2022. arXiv:2207.14406 [cs].

[13] Shagufta Mehnaz, Sayanton V. Dibbo, Ehsanul Kabir, Ninghui Li, and Elisa Bertino. Are your sensitive attributes private? novel model inversion attribute inference attacks on classification models, 2022.

[14] Mikel Hernandez, Gorka Epelde, Ane Alberdi, Rodrigo Cilla, and Debbie Rankin. Standardised metrics and methods for synthetic tabular data evaluation. 09 2021.

[15] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.

[16] Cynthia Dwork. Differential privacy. In Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener, editors, *Automata, Languages and Programming*, pages 1–12, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[17] Shiva P. Kasiviswanathan and Adam Smith. On the "semantics" of differential privacy: A bayesian formulation. *Journal of Privacy and Confidentiality*, 6(1), June 2014.

[18] David McClure and Jerome Reiter. Differential privacy and statistical disclosure risk measures: An investigation with binary synthetic data. *Transactions on Data Privacy*, 5:535–552, 12 2012.

[19] Víctor Elvira and Luca Martino. Advances in importance sampling, 2022.

[20] Damien Desfontaines. A list of real-world uses of differential privacy. https://desfontain.es/blog/real-world-differential-privacy.html, 10 2021. Ted is writing things (personal blog).

A APPENDIX

In this Section we include figures that were not included in the report due to their lesser importance as the figures included.

A.1 Full ADR Model

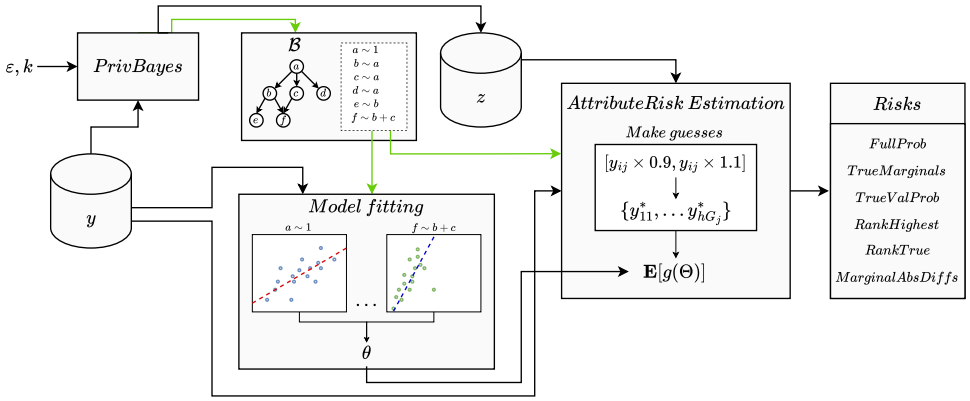

Fig. 19. Figure showcasing the whole process of measuring the ADR as well as the modification done to the output from PrivBayes.

Here, we showcase the full model used to measure the ADR for synthetic dataset generated by PrivBayes. This Figure capture the same process as showcased when combining Figure 4 and Figure 5.

A.2 Density Plots for Continuous Attributes in the Different Datasets

In this section, we include figures showcasing the distributions of continuous attributes of the three different datasets, as well as the distributions where we have injected outliers y_{IN} and y_{OUT}.

A.2.1 Density Plots for CEData.

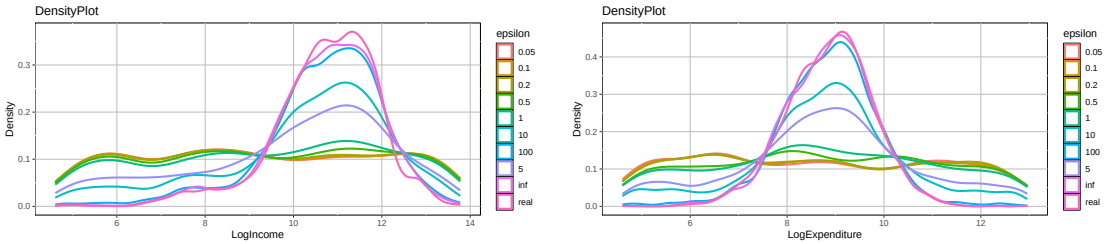

Fig. 20. Figure showcasing the distribution $\{LogIncome, LogExpenditure\}$ from CEData for the real and synthetic datasets given different ε.

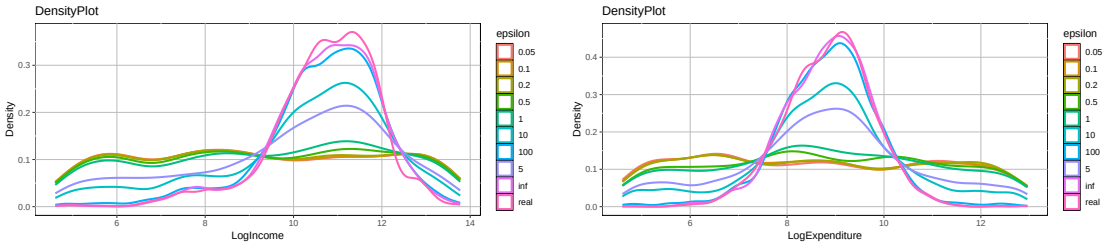

Fig. 21. Figure showcasing the distribution $\{LogIncome, LogExpenditure\}$ from CEData for the real and synthetic datasets given different ε, where we injected y_{IN}.

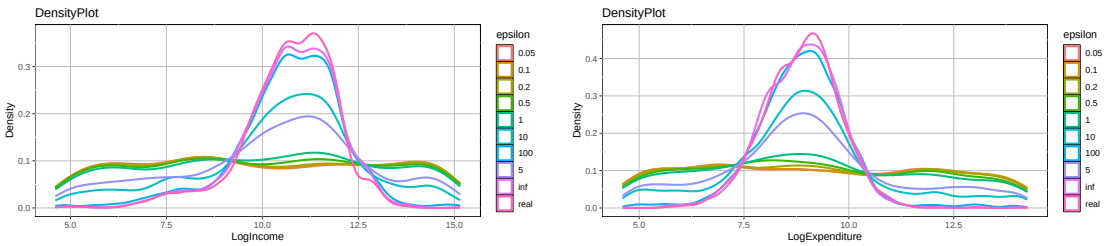

Fig. 22. Figure showcasing the distribution $\{LogIncome, LogExpenditure\}$ from CEData for the real and synthetic datasets given different ε, where we injected y_{OUT}.

A.2.2 *Density Plots for DPHP.*

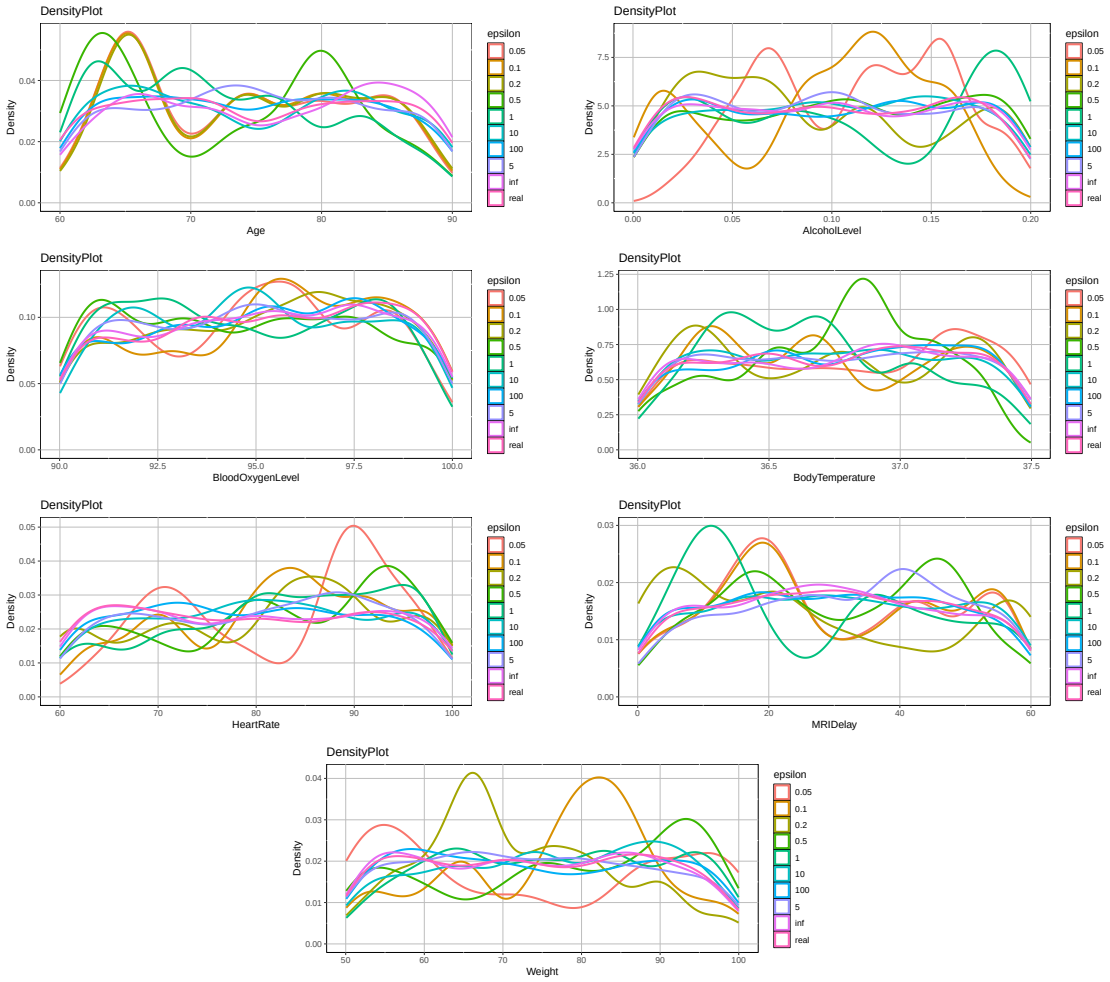

Fig. 23. Figure showcasing the distribution $\{AlcoholLevel, HeartRate, BloodOxygenLevel, BodyTemperature, Weight, MRI\_Delay, Age\}$ from DPHP for the real and synthetic datasets given different ε.

587

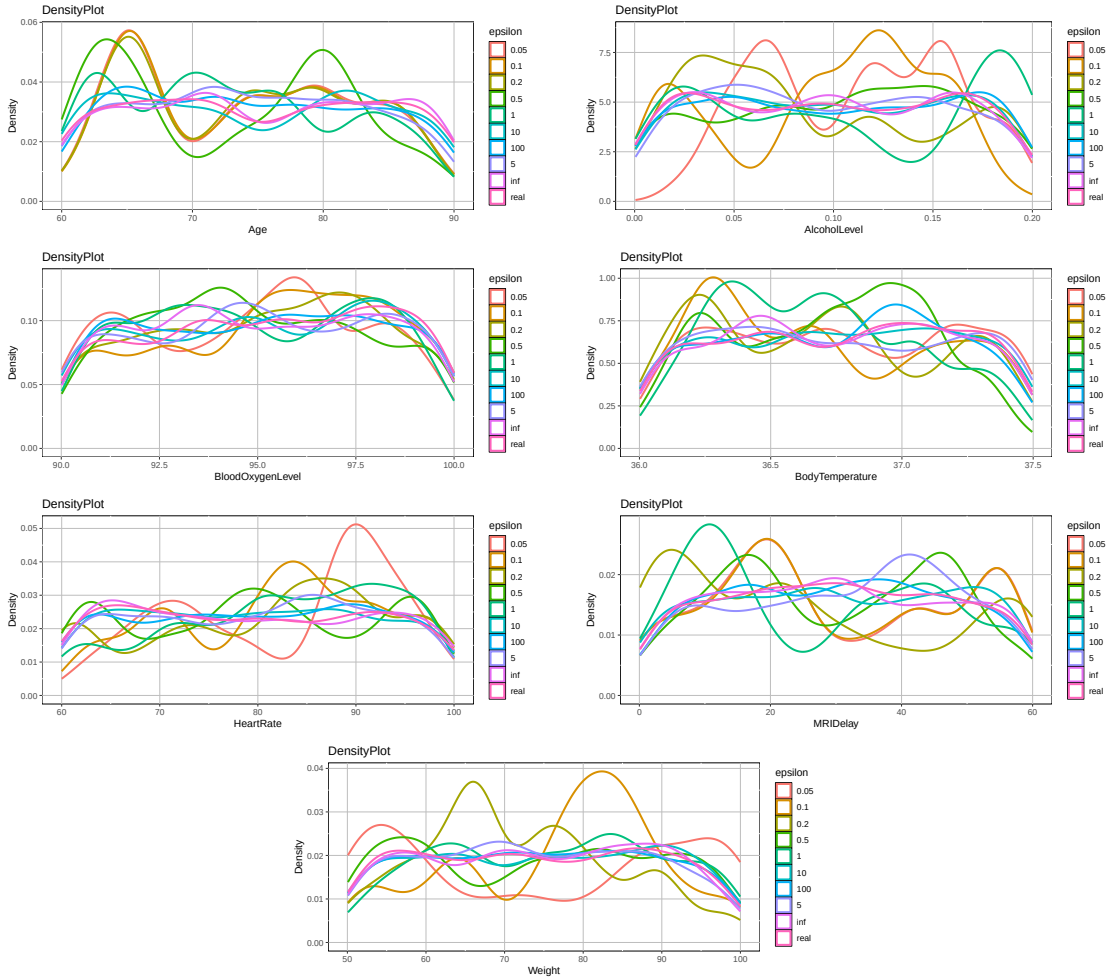

Fig. 24. Figure showcasing the distribution $\{AlcoholLevel, HeartRate, BloodOxygenLevel, BodyTemperature, Weight, MRI\_Delay, Age\}$ from DPHP for the real and synthetic datasets given different ε, where we injected the individual y_{IN}.

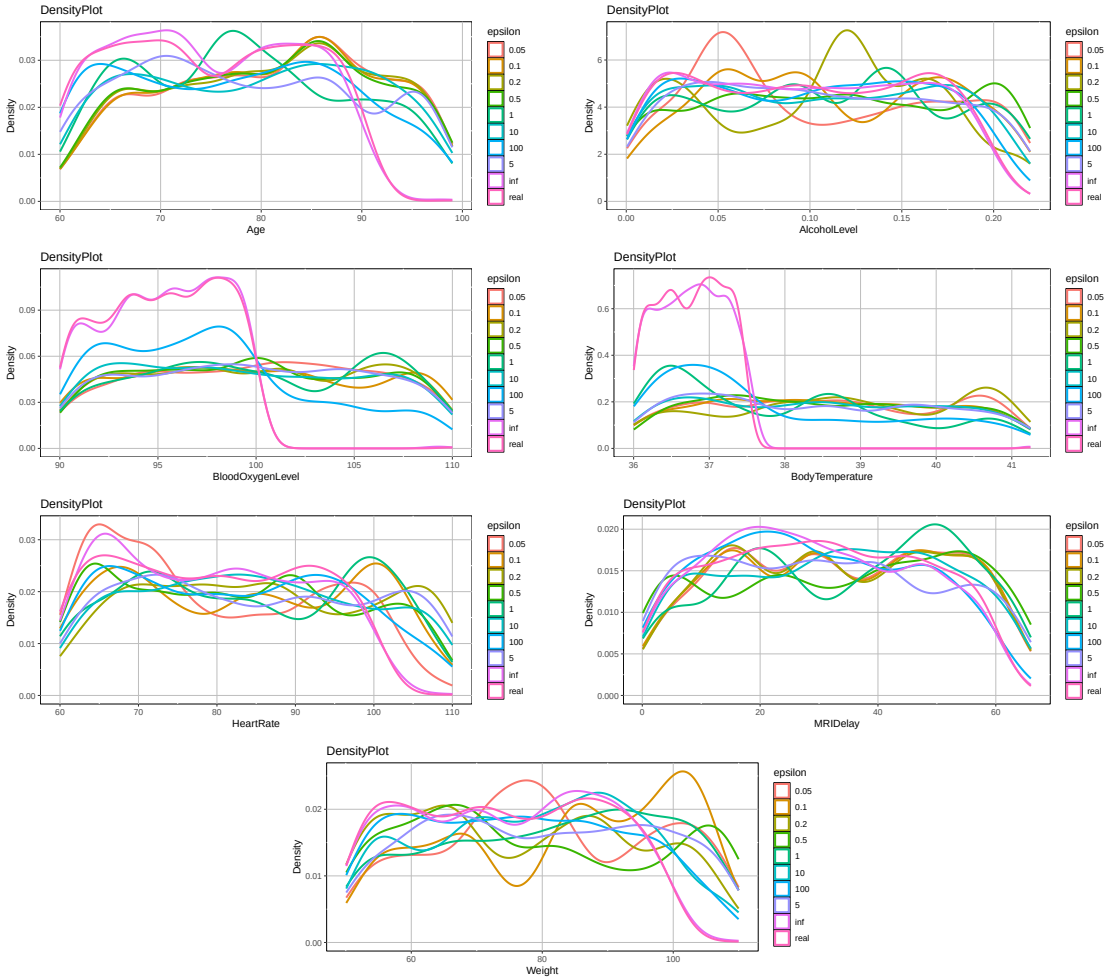

Fig. 25. Figure showcasing the distribution $\{AlcoholLevel, HeartRate, BloodOxygenLevel, BodyTemperature, Weight, MRI\_Delay, Age\}$ from DPHP for the real and synthetic datasets given different ε, where we injected the individual y_{OUT}.

A.2.3 Density Plots for OL.

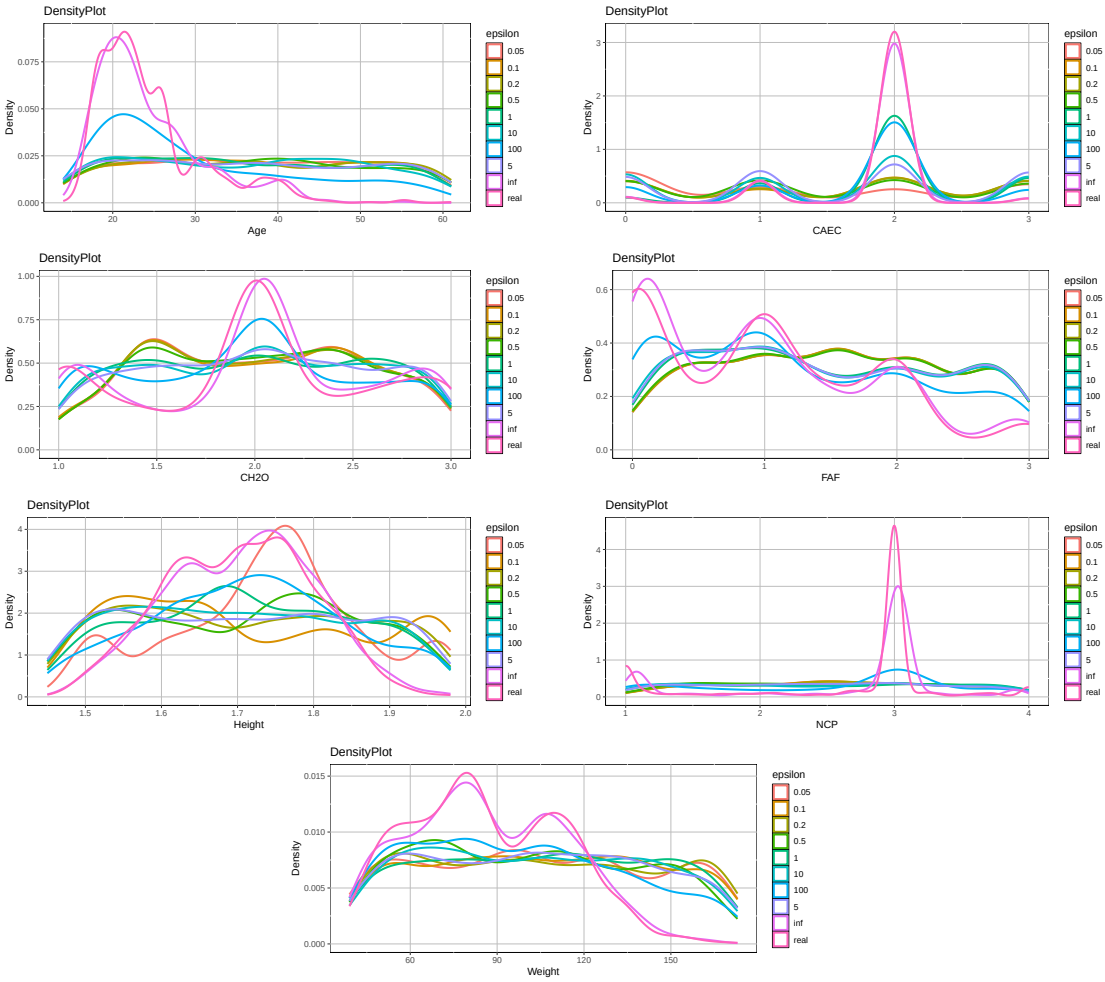

Fig. 26. Figure showcasing the distribution $\{Age, CAEC, CH2O, FAF, Height, NCP, Weight\}$ from OL for the real and synthetic datasets given different ε.

588

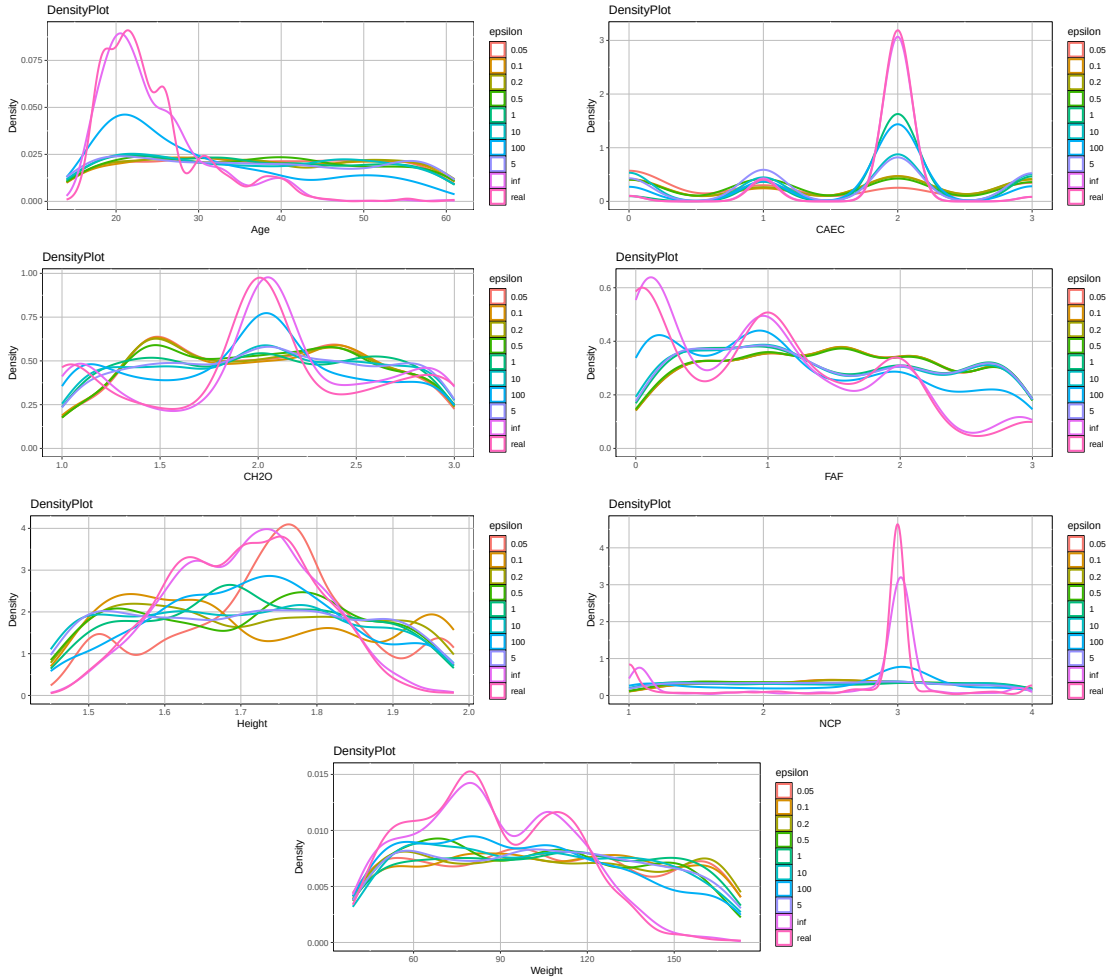

Fig. 27. Figure showcasing the distribution $\{Age, CAEC, CH2O, FAF, Height, NCP, Weight\}$ from OL for the real and synthetic datasets given different ε, where we injected the individual y_{IN}.

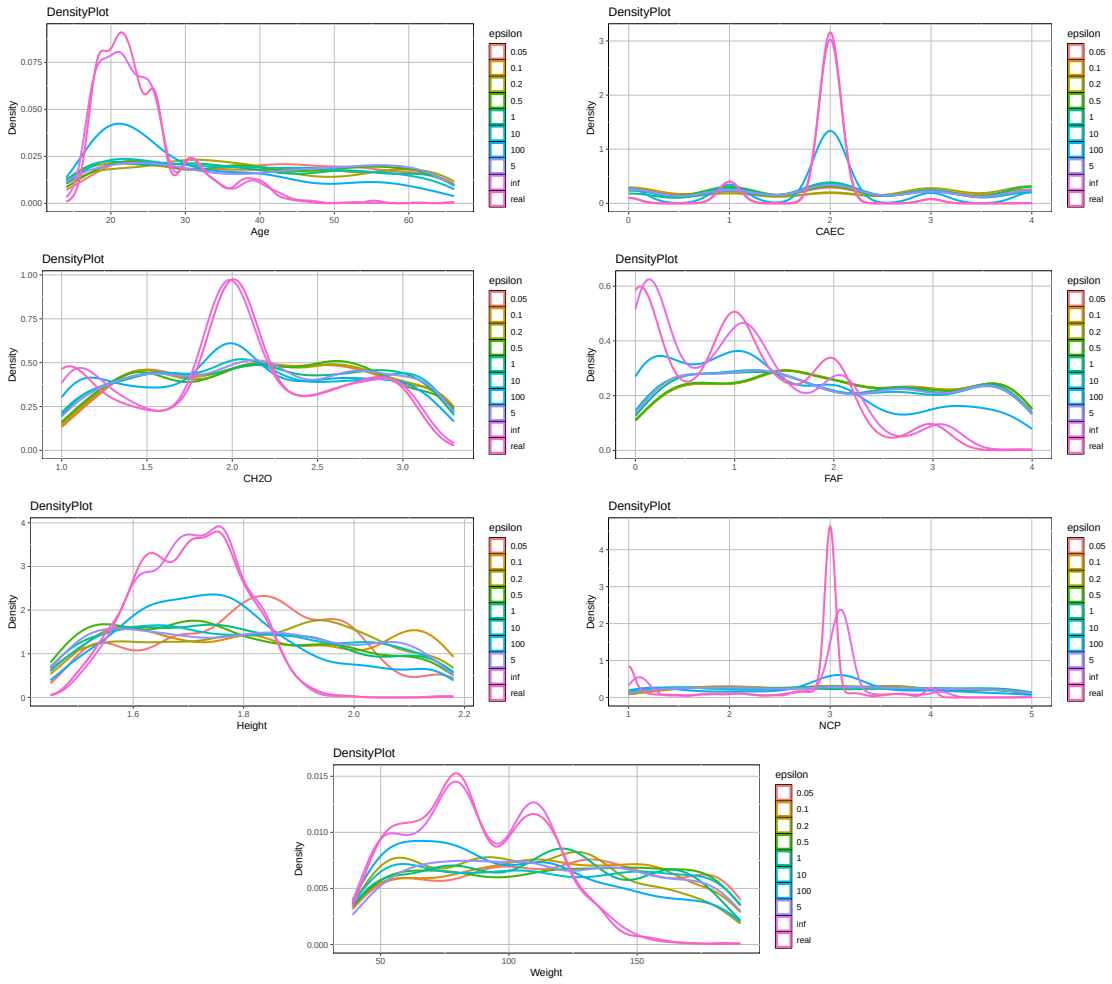

Fig. 28. Figure showcasing the distribution $\{Age, CAEC, CH2O, FAF, Height, NCP, Weight\}$ from OL for the real and synthetic datasets given different ε, where we injected the individual y_{OUT}.

A.3 Density of Probability of Guessing Correctly Plots for Continuous Attributes in the Different Datasets

In this section, we include figures showcasing the density of the probability of guessing attribute values correctly for continuous attributes of the three different datasets as well as the distributions where we have inject outliers y_{IN} and y_{OUT}.

A.3.1 Probability of Guessing Correctly Plots for CEData.

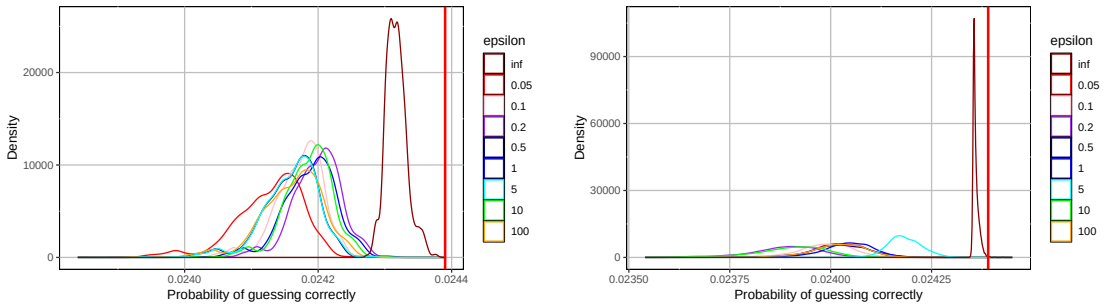

Fig. 29. Figure showcasing the density of our posterior probability of guessing correct for $\{LogIncome, LogExpenditure\}$ from CEData for the real and synthetic datasets given different ε.

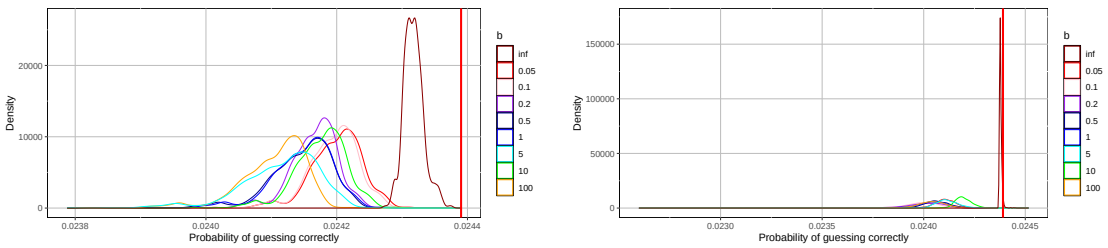

Fig. 30. Figure showcasing the density of our posterior probability of guessing correct for $\{LogIncome, LogExpenditure\}$ from CEData for the real and synthetic datasets given different ε, where we injected y_{IN}.

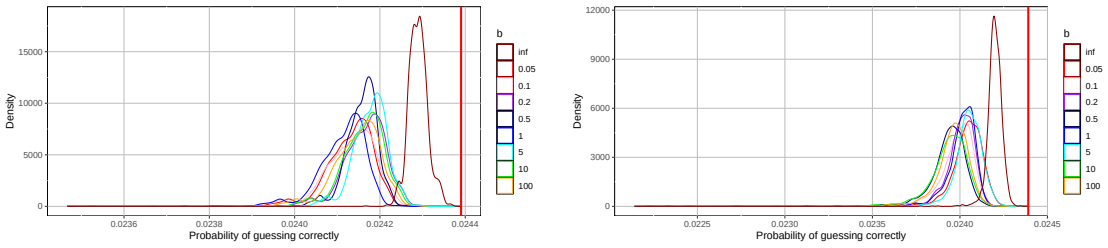

Fig. 31. Figure showcasing the density of our posterior probability of guessing correct for $\{LogIncome, LogExpenditure\}$ from CEData for the real and synthetic datasets given different ε, where we injected y_{OUT}.

A.3.2 Probability of Guessing Correctly Plots for DPHP.

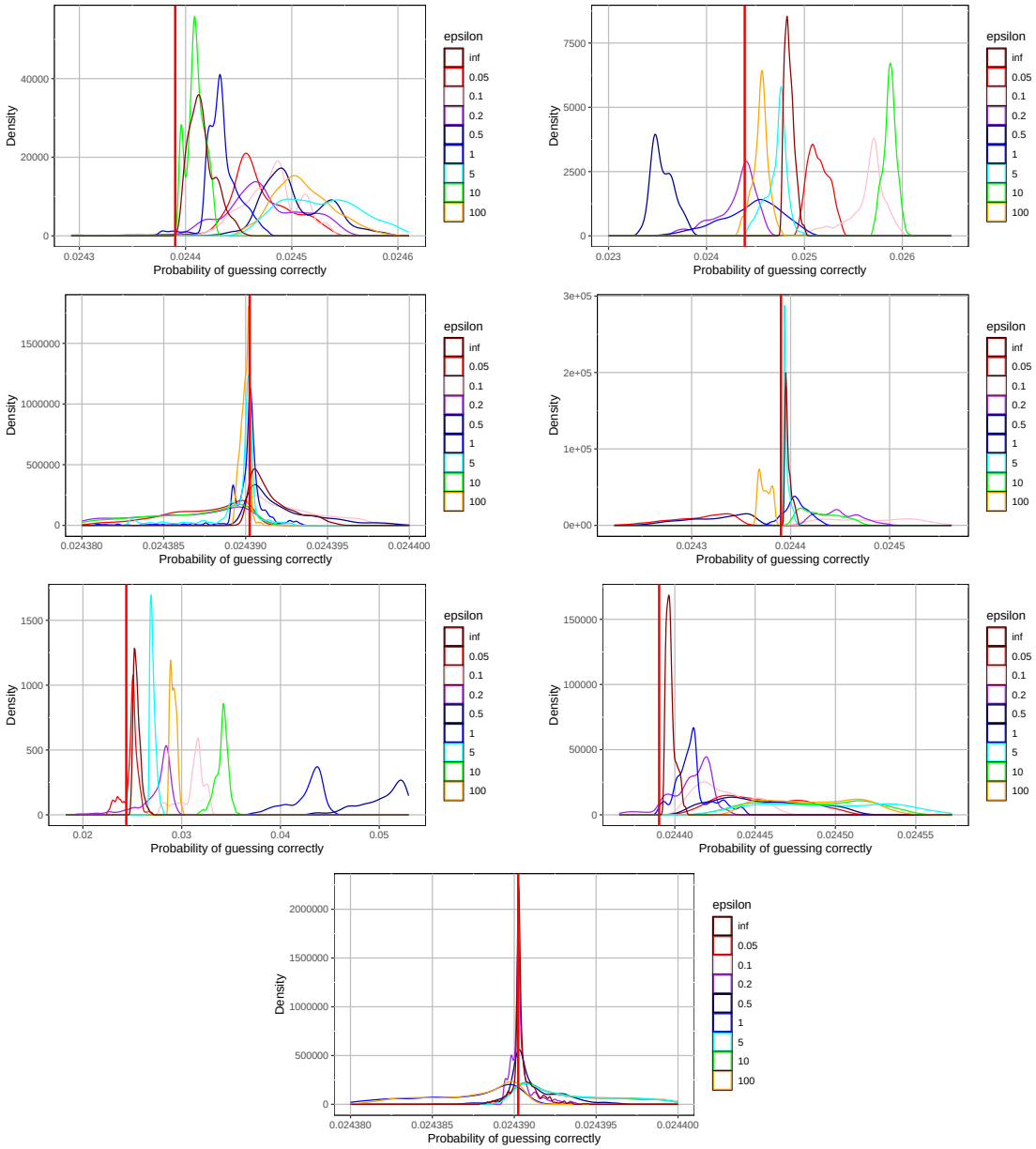

Fig. 32. Figure showcasing the density of our posterior probability of guessing correct for $\{AlcoholLevel, HeartRate, BloodOxygenLevel, BodyTemperature, Weight, MRI\_Delay, Age\}$ from DPHP for the real and synthetic datasets given different ε.

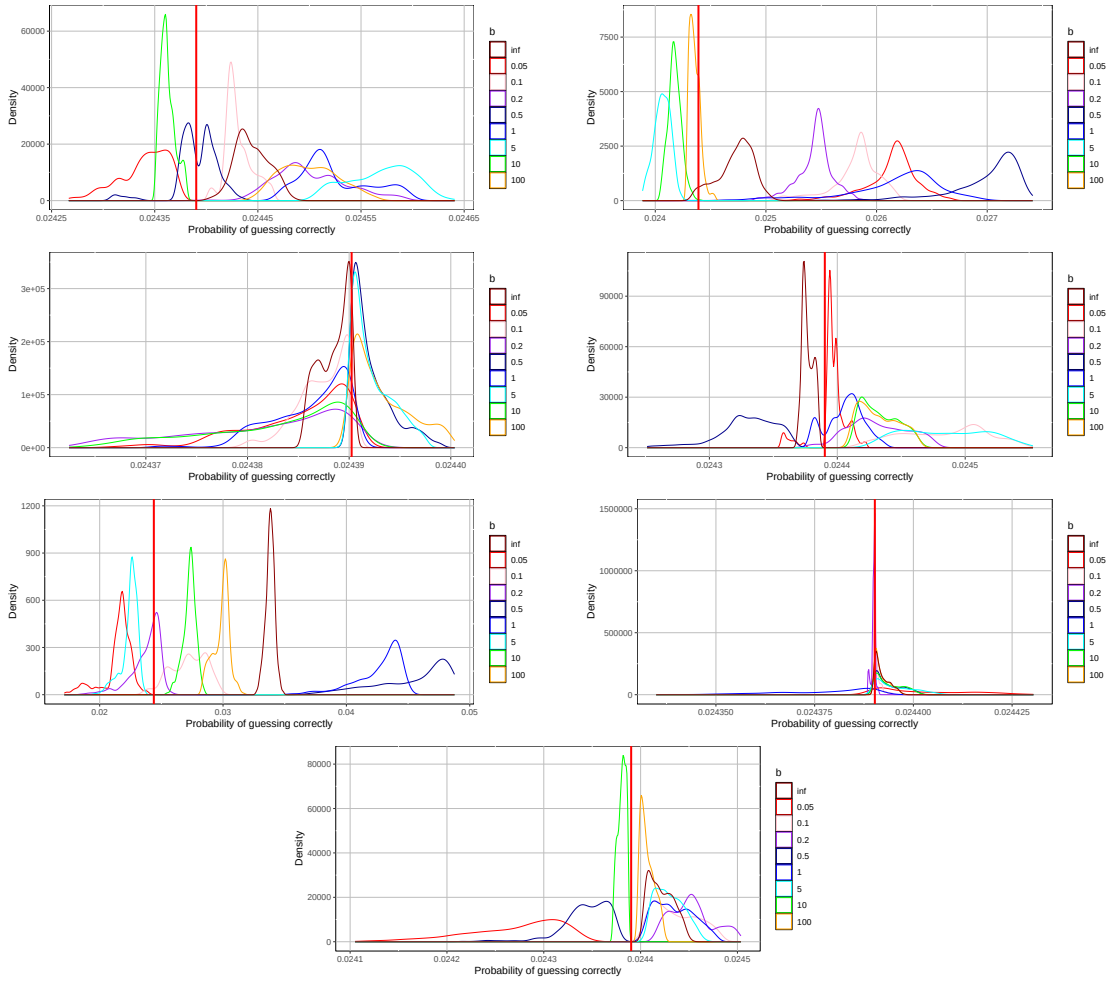

Fig. 33. Figure showcasing the density of our posterior probability of guessing correct for $\{AlcoholLevel, HeartRate, BloodOxygenLevel, BodyTemperature, Weight, MRI\_Delay, Age\}$ from DPHP for the real and synthetic datasets given different ε, where we injected the individual y_{IN}.

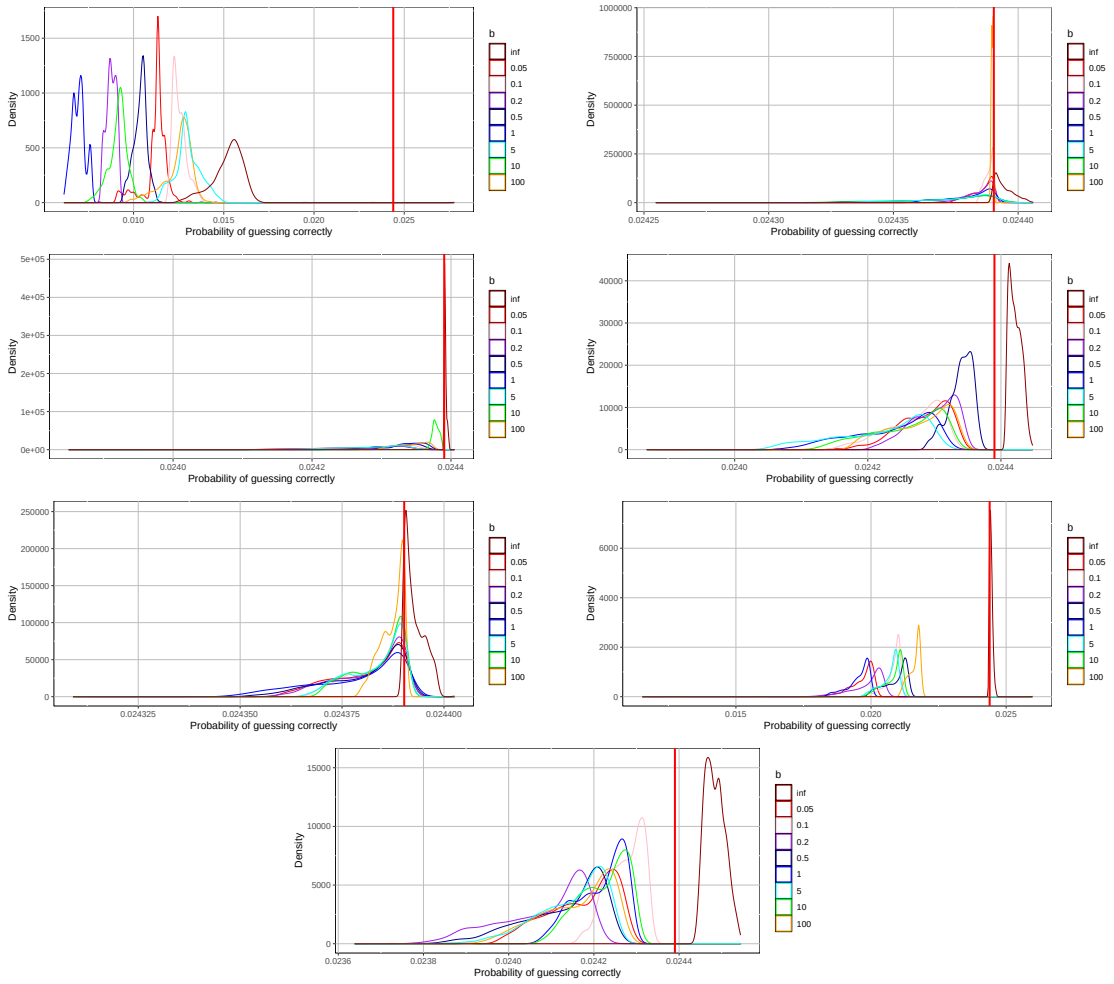

Fig. 34. Figure showcasing the density of our posterior probability of guessing correct for $\{AlcoholLevel, HeartRate, BloodOxygenLevel, BodyTemperature, Weight, MRI\_Delay, Age\}$ from DPHP for the real and synthetic datasets given different ε, where we injected the individual y_{OUT}.

A.3.3 Probability of Guessing Correctly Plots for OL.

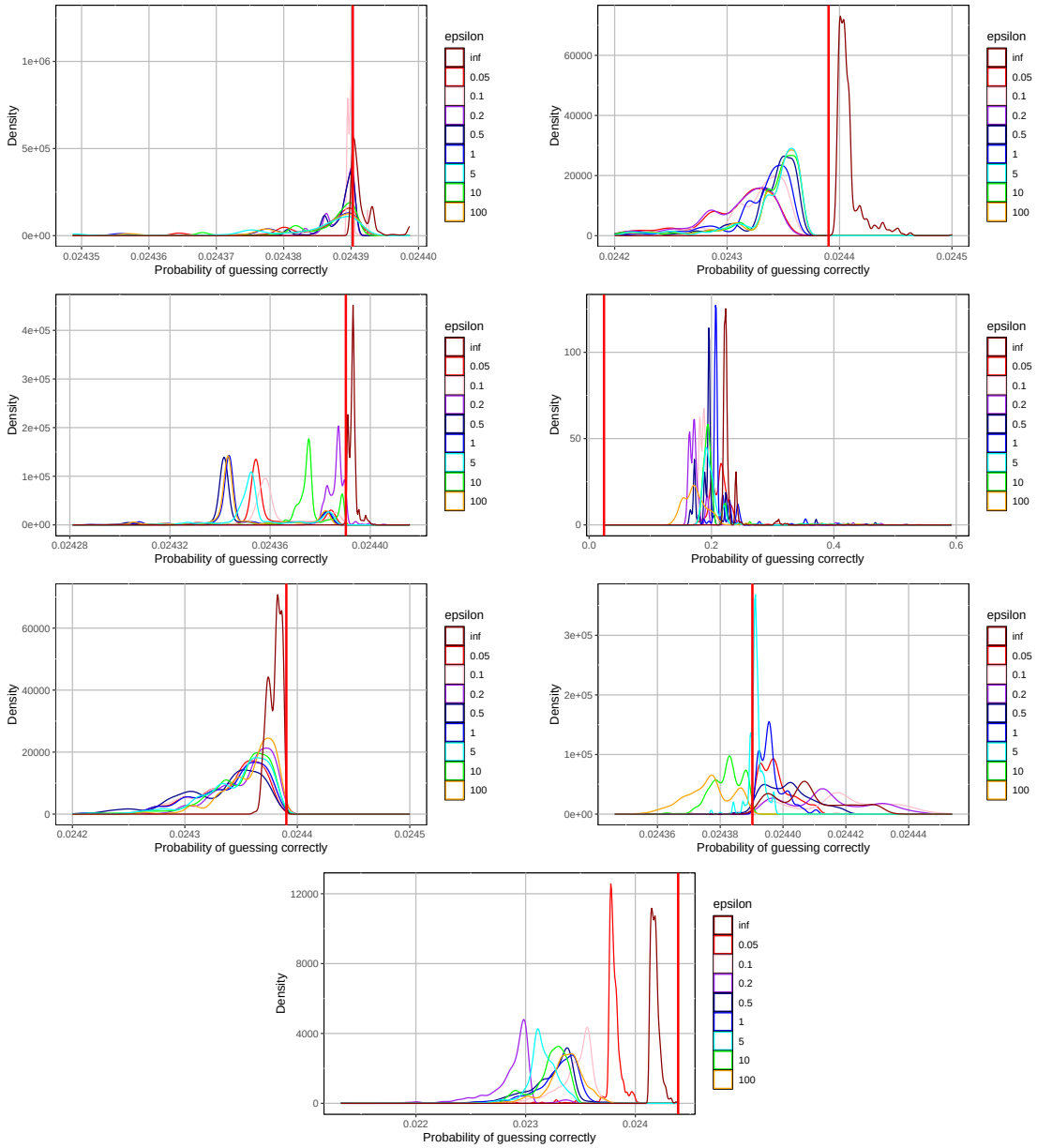

Fig. 35. Figure showcasing the density of our posterior probability of guessing correct for $\{Age, CAEC, CH2O, FAF, Height, NCP, Weight\}$ from OL for the real and synthetic datasets given different ε.

595

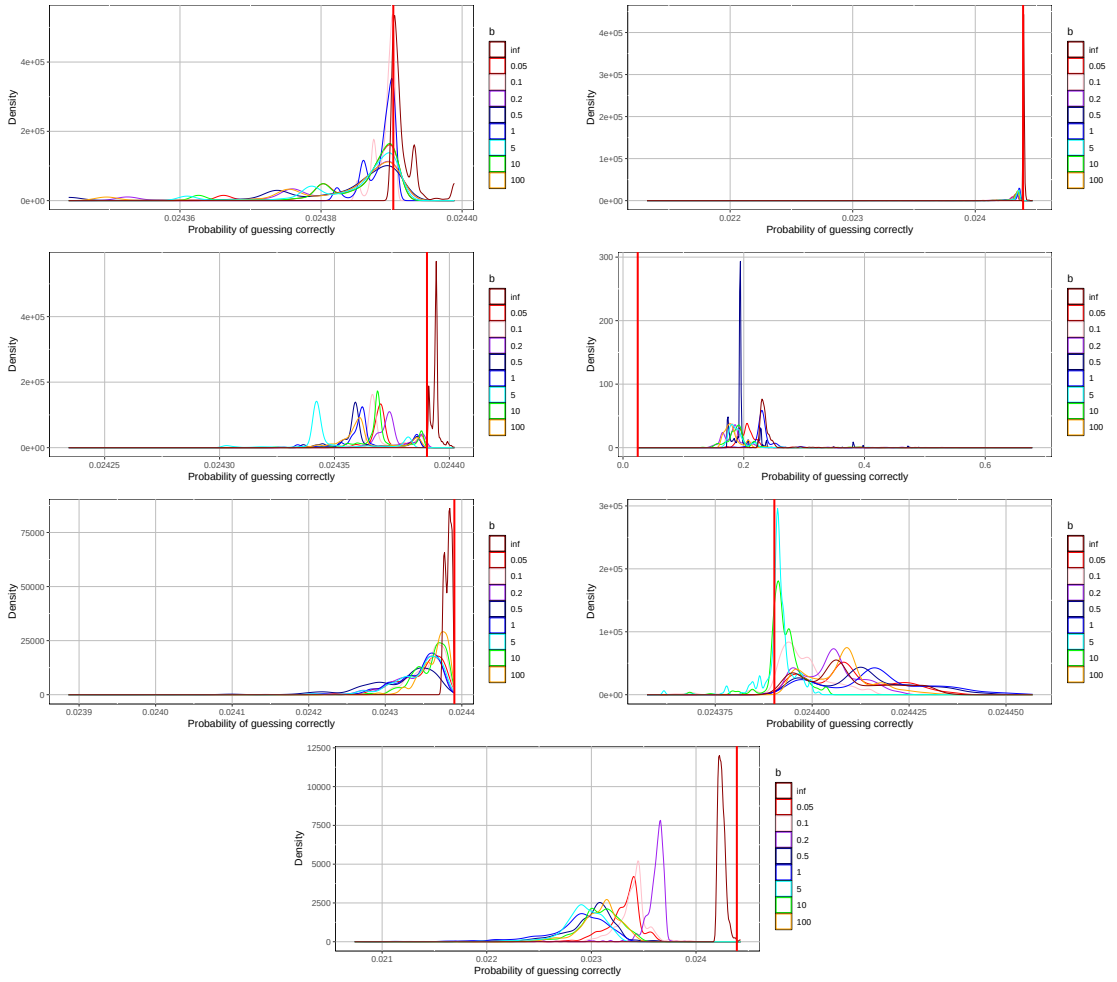

Fig. 36. Figure showcasing the density of our posterior probability of guessing correct for $\{Age, CAEC, CH2O, FAF, Height, NCP, Weight\}$ from OL for the real and synthetic datasets given different ε, where we injected the individual y_{IN}.

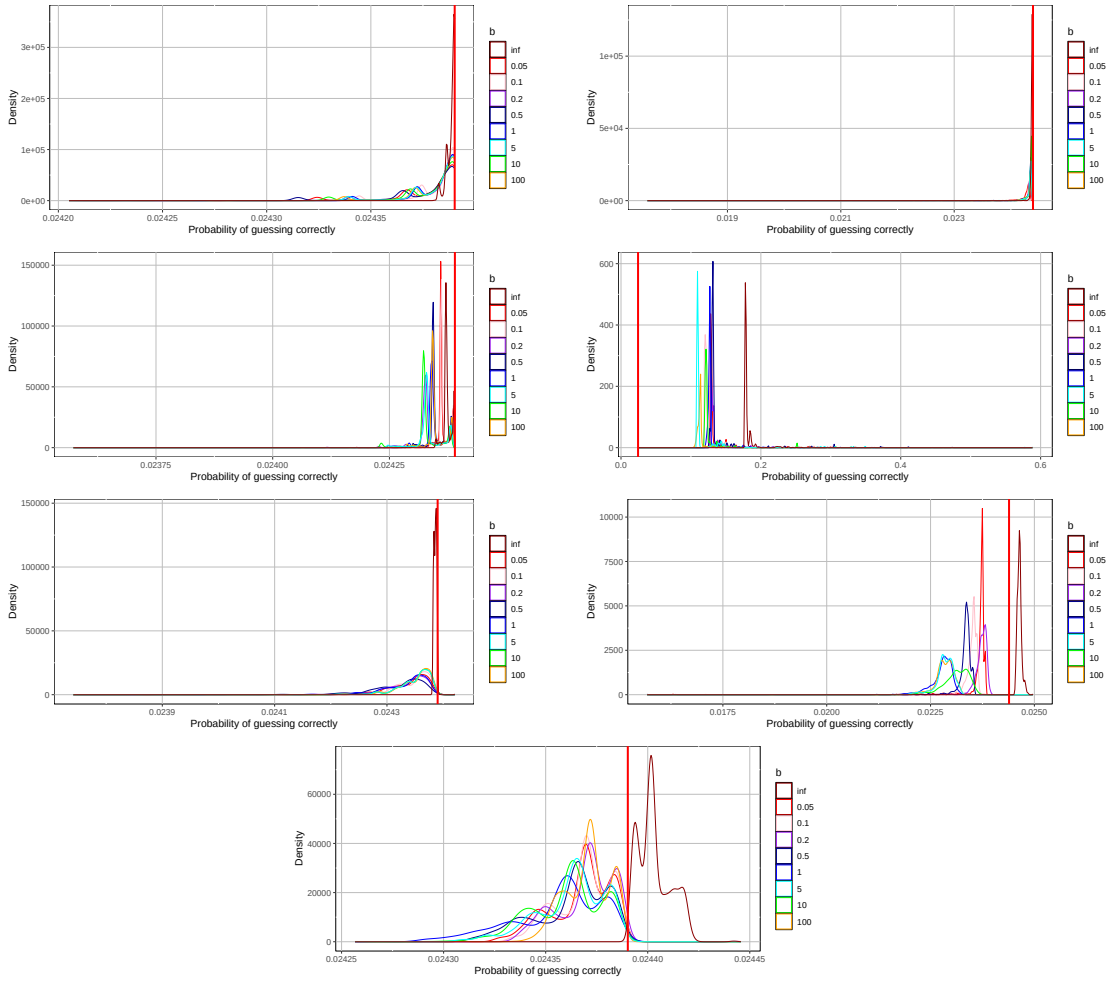

Fig. 37. Figure showcasing the density of our posterior probability of guessing correct for $\{Age, CAEC, CH2O, FAF, Height, NCP, Weight\}$ from OL for the real and synthetic datasets given different ε, where we injected the individual y_{OUT}.