This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.3036422, IEEE Journal of Biomedical and Health Informatics

Mosaic Privacy-preserving Mechanisms for Healthcare Analytics

Alexander Krall, Daniel Finke, Hui Yang*, Senior Member

***Abstract*—The Internet of Things (IoT) has propelled the evolution of medical sensing technologies to greater heights. Thus, traditional health systems have been transformed into new data-rich environments. This provides an unprecedented opportunity to develop new analytical methods and tools towards a new paradigm of smart and interconnected health systems. Nevertheless, there are risks pertinent to increasing levels of system connectivity and data accessibility. Cyber-attacks become more prevalent and complex, leading to greater likelihood of data breaches. These events bring sudden disruptions to routine operations and cause the loss of billions of dollars. Adversaries often attempt to leverage models to learn a target's sensitive attributes or extrapolate its inclusion within a database. As healthcare systems are critical to improving the wellbeing of our society, there is an urgent need to protect the privacy of patients and minimize the risk of model inversion attacks. This paper presents a new approach, named Mosaic Gradient Perturbation (MGP), to preserve privacy in the framework of predictive modeling, which meets the requirement of differential privacy while mitigating the risk of model inversion. MGP is flexible in fine-tuning the trade-offs between model performance and attack accuracy while being highly scalable for large-scale computing. Experimental results show that the proposed MGP method improves upon traditional gradient perturbation to mitigate the risk of model inversion while offering greater preservation of model accuracy. The MGP technique shows strong potential to circumvent paramount costs due to privacy breaches while maintaining the quality of existing decision-support systems, thereby ushering in a privacy-preserving smart health system.**

Key Words: Differential privacy, model inversion attack, Internet of Things, predictive modeling, machine learning, health informatics, mosaic gradient perturbation.

I. Introduction

The advent of the Internet of Things (IoT) has enabled a multitude of data-driven innovations in health care. IoT technology equips a variety of medical things (e.g., wearable sensors, computer servers, medical devices, and human subjects) with internet access and computing power, resulting in a new cyber-physical infrastructure. This new paradigm connects the medical things, databases, and digital clouds in an integrated framework [1]. Healthcare industries are investing in the potential of IoT platforms for the improved delivery of care solutions. The projected size of the global market for wearable devices in the healthcare sector is projected to exceed $17.8 billion by 2021 [2]. As such, traditional health systems have been transformed into new data-rich environments. This provides an unprecedented opportunity to develop new analytical methods and tools to realize a new paradigm of smart and interconnected health systems. For example, Yang *et al.* [3] leveraged IoT sensing alongside a stochastic network model to detect disease patterns present in electrocardiogram signals. The network framework was structured to be highly parallelized to accommodate big data. Liu *et al.* [4] integrated intelligent hardware with deep learning and a mobile terminal to implement an IoT platform for smart dental health. The IoT framework is capable of detecting seven different dental diseases from clinical images with high degrees of sensitivity and specificity. As such, this new IoT technology reduced the mean diagnosis time and increased the number of treated patients. In addition, Zhou *et al.* [5] designed an IoT-enabled telerobotic architecture to support smart health at home. The system incorporates human-motion sensing with workspace mapping and path planning to facilitate a dual-arm robot's completion of various tasks.

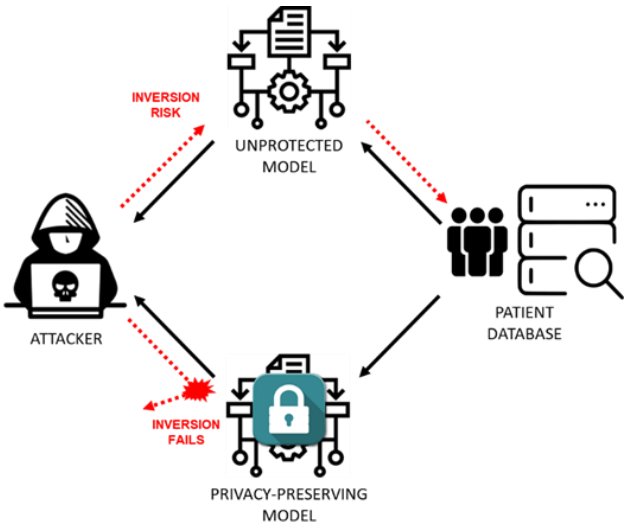

Figure 1. Model inversion attack on a patient database on an unprotected model versus a protected model.

Modern industries increasingly invest in IoT sensors and devices for smart health, which leads to the proliferation of large amounts of data. Although significant works have been performed to address the effectiveness and efficiency of data analytics, little has been done in the realm of IoT privacy in the context of data analytics for the intensive care units (ICU). In the literature, Cai and Kou [6] proposed an encryption and

*This research is supported in part by NSF I/UCRC Center for Healthcare Organization Transformation (CHOT), NSF I/UCRC award #1624727.

A. Krall is with the Complex Systems Monitoring, Modeling and Control lab, The Pennsylvania State University, University Park, PA, 16802 USA (e-mail: auk999@psu.edu).

D. Finke is with the Applied Research Lab, University Park, PA, 16804 USA (e-mail: daf903@psu.edu).

*H. Yang is with the Complex Systems Monitoring, Modeling and Control lab, The Pennsylvania State University, University Park, PA, 16802 USA (e-mail: huy25@psu.edu).

recovery algorithm that allows for distributed statistical inference of encrypted data while preserving privacy, even in the presence of cyberattacks. Nonetheless, the sheer quantity of data will inevitably expose patients to increased risks, especially as the sophistication of data exfiltration attacks intensify. Privacy breaches interrupt standard operations within an institution, which can become costly depending on the level of severity. Such disruptions are projected to cost the industry $300 billion annually [7]. Furthermore, industry-standard data anonymization techniques do not provide a substantial level of protection to the privacy of patients while also guaranteeing the effectiveness of data analytics [8].

Differential privacy provides a viable solution to address the issue of data breaches and realize privacy-preserving data analytics for smart health. A differentially private algorithm ensures that one's participation in a dataset, or lack thereof, will not be disclosed [8]. Nonetheless, statistical information obtained from such a private algorithm is liable to be exploited through the process of model inversion [9]. As shown in Figure 1, unprotected models may unintentionally reveal sensitive information about a patient included within a database. By contrast, a privacy-preserving model should be resistant to model inversion attacks.

This paper presents a new privacy-preserving technique for predictive healthcare analytics that mitigates the risk of model inversion while managing model accuracy tradeoffs. Specifically, our contributions are summarized as follows:

1) We develop a Mosaic Gradient Perturbation (MGP) technique, that can preserve differential privacy while giving stronger guarantees against model inversion attacks. The algorithm perturbs the portions of the objective function's gradient associated with sensitive attributes more strongly than those associated with non-sensitive attributes.
2) We conduct a comparison experiment to benchmark the new MGP technique against standard Gradient Perturbation (GP) in terms of both model accuracy and attack robustness.
3) A distributed version of the MGP algorithm is developed to allow for distributed computing and parallel processing of datasets while preserving data privacy.
4) We evaluate and compare the computational efficiency between distributed-processing and serial-computing versions of MGP algorithms.

These contributions enable the effective implementation of differential privacy to the cyber-physical healthcare landscape, which help deter significant costs due to privacy breaches. The privacy-preserving algorithms guarantee that the quality of decision-support systems is maintained. Furthermore, the proposed MGP technique enables distributed processing, which provides a scalable tool for privacy-preserving analytics.

The remainder of this paper is organized as follows: Section II provides a literature review into the impetus for utilizing differential privacy in the IoT setting by identifying gaps in the current research. Section III expands upon the notion of differential privacy and model inversion. The new MGP algorithms are developed in this section. Section IV discusses the experimental design pertinent to an intensive care unit dataset. Section V includes diagrams of the model accuracies and attack robustness observed for each technique as privacy parameters vary. Additionally, the comparison of computational efficiency between the distributed and serial versions of MGP are shown in this section. Section VI includes a discussion of benefits and limitations of MGP and the direction for future work. Section VII concludes this study with an overview of privacy-preserving data analytics for smart and interconnected healthcare systems.

II. Research Background

There are a number of disturbing headlines released that highlight the vulnerabilities present in anonymized data. A couple examples of these include "Matching Known Patients to Health Records in Washington State Data" [10] and "Identifying Participants in the Personal Genome Project by Name" [11]. Balancing the tradeoffs between model privacy and utility is difficult when relying solely on data anonymization. When data resolution and dimensionality are high, rows in a database are essentially unique. Therefore, linking a set of observations to an individual becomes a much simpler matter. Advances in IoT sensing will also result in an increase in data diversity and resolution, which are collected across a multitude of devices. Because the quality and quantity of collected data are expanding, one can surmise that truly private anonymization will become an intractable feat as technology progresses. Reducing dimensionality or resolution has the unfortunate consequence of removing information and contributes to the loss of predictive accuracy. Thus, new privacy techniques are urgently needed such that one can maintain privacy while capitalizing on the newfound ability to leverage large amounts of IoT data.

The intensive care unit (ICU) is one such data-rich environment due to IoT-based transformations. The ICU treats critically ill patients and puts them under the constant observation of physicians and monitoring equipment. Due to the circumstances in which a patient is admitted to the ICU, a primary concern is a patient's risk of mortality. Countermeasures can be taken to circumvent mortality, which will be dependent on a patient's risk level. Determination of this risk has traditionally been done through a variety of methods, including: Acute Physiology and Chronic Health Evaluation (APACHE), the Simplified Acute Physiology Score (SAPS), the Mortality Probability Model (MPM), and the Sequential Organ Failure Assessment (SOFA). These heuristics utilize their associated scores to fit logistic regression models for the prediction of mortality. Despite their common use in the ICU, these traditional methods of assessing mortality risk face key issues with variable heterogeneity, patient heterogeneity, and time asynchronization. In order to address these shortcomings, Chen *et al.* [12] developed a machine learning model to predict the mortality risk and stratify the patients for optimal resource allocations. Despite

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.3036422, IEEE Journal of Biomedical and Health Informatics

the effectiveness in leveraging data to produce highly accurate models in healthcare analytics, one should not be ignorant to the implications on privacy. Predictive models that are accurate have the unfortunate consequence of becoming more easily exploited for model inversion attacks if disclosed.

Differential privacy was developed to address the exploitability of models. Chaudhuri *et al.* [13] developed two algorithms that achieve differential privacy for a logistic regression model. The first algorithm works by perturbing the model's output regression coefficients whereas the second one works by perturbing the model's entire objective function. Zhang *et al.* [14] produced a variation of objective perturbation called the functional mechanism. Under this algorithm, the data coefficients to regression parameters are perturbed. In light of the possibility of model inversion attacks, Wang *et al.* [15] expanded the functional mechanism technique where coefficients corresponding to sensitive attributes are perturbed more strongly than those associated with non-sensitive attributes. This amended technique is known as the sensitive mechanism. Output and objective perturbation may be acceptable to employ from a centralized computing perspective. However, in the era of high-volume data, the capability of distributed processing is preferred by the means of gradient-based methods. Nonetheless, the functional mechanism (and by extension the sensitive mechanism) suffers greatly from accuracy loss as data dimensionality increases [16], and is limited in its ability to handle any form of distributed processing.

Also, genomics is a field that deals with intimate personal data and has extensively leveraged differential privacy. For example, Raisaro *et al.* [17] applied homomorphic encryption alongside differential privacy to enhance the privacy guarantees of the Informatics for Integrating Biology and Bedside framework without significantly compromising computational expediency. Nonetheless, Fredrikson *et al.* [9] show that application of differential privacy to genomics must contend with the issue of dependent tuples. Correlations between database entries may weaken the privacy guarantees of a differentially private algorithm. Almadhoun *et al.* [18] deals with this issue by adjusting the global sensitivity of the query. The adjustment utilizes the notion of a leaked information ratio in the presence of multiple privacy budget values. Experimental results show that this adjustment can mitigate inversion risk.

III. Differential Privacy

In the context of predictive modeling, a database D will contain n tuples, each with d input variables $\mathbf{x}_i = (x_{i1}, \dots, x_{id})$ and one response y_i. The domain of $\mathbf{x}_i$ is assumed to be confined to the L^1 norm, where $\|\mathbf{x}_i\|_1 \leq 1$. In this investigation, assume that the response's domain is binary, $y_i \in \{-1,1\}$. The machine learning task is to train a logistic regression model and release its parameters $\omega = (\omega_1, \dots, \omega_d)$.

3.1 Conceptual Foundations

Under the differential privacy, one's inclusion within a dataset should make no statistical difference in an algorithm's output. Therefore, two databases that only differ by a single record should produce statistically similar results when running a private algorithm [8].

Definition 1. *A randomized function A gives ϵ-differential privacy if for all datasets D and D' differing by at most one row and for all $\xi \subseteq Range(A)$,*

$$\Pr\{A(D) \in \xi\} \leq e^{\epsilon} \Pr\{A(D') \in \xi\} \quad (1)$$

Privacy parameter ϵ controls the degree of difference allowed between output distributions derived from applying an algorithm $A(\cdot)$ onto databases D and D'. Figure 2 shows a visualization of Definition 1's premise.

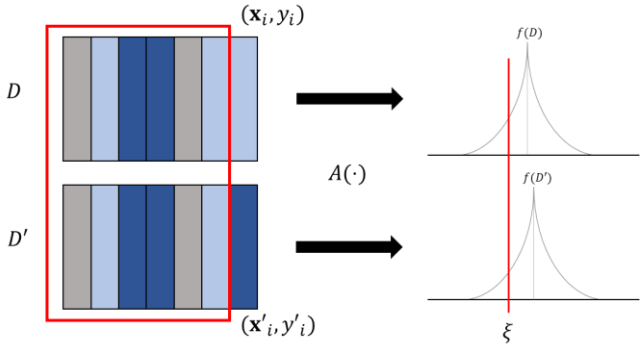

Figure 2. The output of a differentially private algorithm when two databases differ by only one row.

The Laplace mechanism is a means to preserve ϵ-differential privacy for any function f. Noise from a Laplace distribution is applied to the function's output,

Theorem 1. *An algorithm A takes as input a dataset D and some $\epsilon > 0$, a query Q with computing function $f: D^n \rightarrow R^d$, and outputs,*

$$A(D) = f(D) + (\Upsilon_1, \dots, \Upsilon_d) \quad (2)$$

where the Υ_k are drawn i.i.d. from $Lap(\Delta/\epsilon)$, and Δ is the global sensitivity of f. The algorithm satisfies ϵ-differential privacy.

Definition 2. *The global sensitivity of a function $f: D^n \rightarrow R^d$ is the following,*

$$\Delta = \max_{D,D' \; s.t. \; D' \in \Gamma(D)} \|f(D) - f(D')\|_1 \quad (3)$$

Note that $\Gamma(D)$ describes the set of all datasets that only differ from D by one row. The magnitude of noise injected into f is partially dependent on Δ, which is known as the global sensitivity of the function. This notion of global sensitivity

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.3036422, IEEE Journal of Biomedical and Health Informatics

defines the maximum possible change in f's output should D be substituted for D'. Privacy budget ϵ also controls the degree of perturbation [8].

3.2 Model Inversion Attack

Inference of one's inclusion in a dataset is a risk mitigated by differential privacy when building a predictive model. However, privacy extends beyond the notion of participation. A private model may still be exploited to a target individual's sensitive attributes in the presence of available auxiliary information. The process of leveraging a model for this purpose is illustrated in Figure 3. Fredrikson *et al.* [9] developed a model inversion algorithm for discretized data to learn a target's genetic markers. The technique offered superior performance in learning sensitive attributes x_s over the baseline approach that utilizes the marginal probabilities of attribute values.

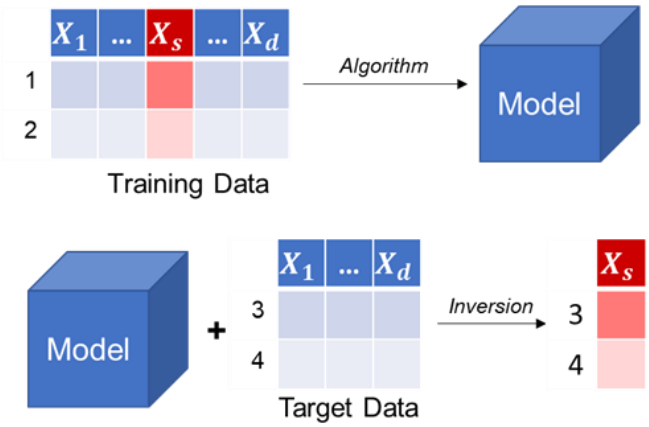

Figure 3. The process of model inversion attack.

The accuracy of model inversion is assessed for continuous attributes. Let a target individual's data be represented by the tuple t_α. It is assumed that the adversary has access to the released regression coefficients ω as well as an estimation of classification probability p_α. Let the sensitive attribute of interest be $x_{\alpha s}$. Likewise, let the regression coefficient associated with $x_{\alpha s}$ be represented by ω_S and the bias term be ω_0. The attacker can learn its value by calculating,

$$x_{\alpha s} = \frac{\text{logit}(p_\alpha) - \omega_0 - \sum_{i \neq s} \omega_i x_{\alpha i}}{\omega_s} \quad (4)$$

3.3 Empirical Risk Minimization

Machine learning models seek to minimize the prediction error given training data. Thus, under the paradigm of empirical risk minimization (ERM), a set of parameters ω are chosen that minimizes the regularized empirical loss function,

$$J(\omega, D) = \frac{1}{n} \sum_{t_i \in D} \ell(\omega, t_i) + \Lambda R(\omega) \quad (5)$$

In (5), ℓ is the loss function, Λ is the regularization parameter, and $R(\cdot)$ is the regularization function. The parameter t_i represents a tuple derived from D at row index i. Differentially private ERM is attained through the strategic injection of noise during particular phases of model training. As shown in Table I, this perturbation can be performed on the model's optimal regression coefficients ω^*, the objective function J, or the objective's gradient ∇J.

TABLE I. PERTURBATION METHODS

Perturbation Type	Noise Injected Into
Output	ω^*
Objective	J
Gradient	∇J

We develop a set of gradient-based methods for privacy-preserving healthcare analytics. Note that Song *et al.* first proposed the GP for differentially private updates [19], which does not discriminate between sensitive and non-sensitive attributes in the perturbation method. In addition, the learning rate is not adaptively adjusted for fast convergence. Algorithm 1 is a newly revised implementation of the GP method with an adaptive learning rate. We propose to develop the MGP method (i.e., Algorithm 2) to make the distinction between sensitive and non-sensitive attributes. Such a distinction was previously used for functional mechanisms and linear regression [15], but little has been done for GP methods and logistic regression. Furthermore, we propose an extended distributed computing version (i.e., Algorithm 3) to handle the predictive modeling of larger datasets. Each of these algorithms utilize an adaptive learning rate for fast convergence.

Algorithm 1 Gradient Perturbation

Input: Data D, parameters $\epsilon, \Lambda, \text{K}, b, \theta$
Output: Approximate noisy minimizer $\bar{\omega}$

Initialize $\omega^{(1)}$, $\tau = 1$, $\kappa = 1$, $\eta_0 = \sqrt{\frac{1}{\Lambda^{1/2}}}$
Let $\tau_0 = \frac{1}{\Lambda \eta_0}$
Distribute D into a set of batches B, each of size b
while $\kappa \leq$ K
  for each $j = 1, \ldots, |\text{B}|$ **do**
    Set $\eta^{(\tau)} = \frac{1}{\Lambda(\tau_0 + \tau - 1)}$
    Set $\Delta^{(\tau)} = \frac{2\theta\eta^{(\tau)}}{b}$
    Draw a vector $\mathbf{z}^{(\tau)} \sim Lap\left(\frac{\Delta^{(\tau)}}{\epsilon}\right)$
    Set $\omega^{(\tau+1)} = \omega^{(\tau)} - \eta^{(\tau)}\left(\nabla J\left(\omega^{(\tau)}, \text{B}_j\right) + \frac{1}{b}\mathbf{z}^{(\tau)}\right)$
    Set $\tau = \tau + 1$
  end for
  Set $\kappa = \kappa + 1$
end while
Let $\bar{\omega} = \omega^{(\tau)}$

Algorithm 1 is initialized with a privacy parameter ϵ, a regularization parameter Λ, the number of epochs K, and a batch size b. At the start of the algorithm, the initial learning rate η_0 is automatically determined based on Λ. The initial

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.3036422, IEEE Journal of Biomedical and Health Informatics

value of ω is randomly generated. The iteration counter τ and epoch counter κ are both set to one. An additional parameter, τ_0, is calculated utilizing η_0 and Λ. This parameter is utilized to compute $\eta^{(\tau)}$ at each iteration. Dataset D is divided into batches of size b prior to the main loop of the algorithm. For a given epoch κ, the algorithm iterates over all batches. Next, the learning rate $\eta^{(\tau)}$ is updated for a particular batch B_j. Next, the global sensitivity $\Delta^{(\tau)}$ is updated accordingly. A random vector $\mathbf{z}^{(\tau)} \sim Lap\left(\Delta^{(\tau)}/\epsilon\right)$ is drawn and then scaled by $1/b$. After scaling the noise vector, it is injected into the gradient ∇J when updating ω. Each time ω is updated, the value of τ is incremented by one. Once all mini batches are used to update ω, the epoch counter κ is also incremented by one. This process continues until the maximum number of epochs has been reached.

We now prove that the process of updating coefficients ω in Algorithm 1 satisfies Theorem 1 and is therefore ϵ-differentially private. The proof shown in Theorem 2 applies the results from Lemma 1 and Corollary 1. For the purposes of all proofs, assume without loss of generality that B_j and B_j' are two neighboring batches that differ in the last row. Let $f\left(\mathrm{B}_j\right) = \eta^{(\tau)} \nabla J\left(\omega^{(\tau)}, \mathrm{B}_j\right)$, representing the change in ω, be the function that is perturbed with Laplace noise.

Lemma 1. *The global sensitivity of gradient-descent weight updates is at most* $\frac{2\eta^{(\tau)}}{b} \max_t \left\| \nabla \ell(\omega^{(\tau)}, t) \right\|_1$.

Proof.

$$
\begin{aligned}
\Delta^{(\tau)} &= \left\| f\left(\mathrm{B}_j\right) - f\left(\mathrm{B}_j'\right) \right\|_1 \\
&= \left\| \eta^{(\tau)} \nabla J\left(\omega_j^{(\tau)}, \mathrm{B}_j\right) - \eta^{(\tau)} \nabla J\left(\omega_j^{(\tau)}, \mathrm{B}_j'\right) \right\|_1 \\
&= \left\| \eta^{(\tau)} \left(\Lambda \nabla R(\omega) + \frac{1}{b} \sum_{t_i \in \mathrm{B}_j} \nabla \ell\left(\omega^{(\tau)}, t_i\right) \right) \right. \\
&\qquad \left. - \eta^{(\tau)} \left(\Lambda \nabla R(\omega) + \frac{1}{b} \sum_{t_i \in \mathrm{B}_j'} \nabla \ell\left(\omega^{(\tau)}, t_i\right) \right) \right\|_1 \\
&= \left\| \frac{\eta^{(\tau)}}{b} \left(\nabla \ell\left(\omega^{(\tau)}, t_b\right) - \nabla \ell\left(\omega^{(\tau)}, t_{b'}\right) \right) \right\|_1 \\
&\leq \frac{\eta^{(\tau)}}{b} \left(\left\| \nabla \ell\left(\omega^{(\tau)}, t_b\right) \right\|_1 + \left\| \nabla \ell\left(\omega^{(\tau)}, t_{b'}\right) \right\|_1 \right) \\
&\leq \frac{2\eta^{(\tau)}}{b} \max_t \left\| \nabla \ell\left(\omega^{(\tau)}, t\right) \right\|_1
\end{aligned}
$$

The first inequality results from the triangle inequality. In the second inequality, t is an arbitrary tuple. ■

Corollary 1. *If* $\|\nabla \ell\|_1 \leq \theta$*, the global sensitivity of gradient-descent weight updates is at most* $\frac{2\theta\eta^{(\tau)}}{b}$.

Theorem 2. *Algorithm 1 satisfies ϵ-differential privacy.*

Proof. We compare the function f in the presence of B and B′ at some arbitrary point g.

$$
\begin{aligned}
\frac{\Pr\{f(\mathrm{B}_j)\}}{\Pr\{f(\mathrm{B}_j')\}} &= \prod_{k=1}^{d} \frac{\exp\left(-\frac{\epsilon}{\Delta^{(\tau)}} \left| f_k(\mathrm{B}_j) - g_k \right|\right)}{\exp\left(-\frac{\epsilon}{\Delta^{(\tau)}} \left| f_k(\mathrm{B}_j') - g_k \right|\right)} \\
&= \prod_{k=1}^{d} \exp\left(\frac{\epsilon\left(\left| f_k(\mathrm{B}_j') - g_k \right| - \left| f_k(\mathrm{B}_j) - g_k \right| \right)}{\Delta^{(\tau)}} \right) \\
&\leq \prod_{k=1}^{d} \exp\left(\frac{\epsilon \left| f_k(\mathrm{B}_j) - f_k(\mathrm{B}_j') \right|}{\Delta^{(\tau)}} \right) \\
&= \exp\left(\frac{\epsilon \left\| f(\mathrm{B}_j) - f(\mathrm{B}_j') \right\|_1}{\Delta^{(\tau)}} \right) \\
&\leq \exp(\epsilon)
\end{aligned}
$$

The first inequality follows from the triangle inequality. The second inequality follows from Corollary 1. ■

3.4 Mosaic Gradient Perturbation

We propose Algorithm 2 to improve the privacy of sensitive features by weakening their correlation with the response variable. Hence, the gradient corresponding to sensitive attributes is perturbed more intensely. Thus, a separate privacy budget is allocated for non-sensitive and sensitive attributes, ϵ_N and ϵ_S. A ratio parameter γ is established such that $\gamma = \epsilon_N/\epsilon_S$ and $0 < \gamma \leq 1$. A smaller γ will inject more noise into sensitive attributes. Ergo, data labels Φ of D are partitioned into the sets Φ_N and Φ_S. Let ω_ϕ corresponds to the coefficient associated with data label $\phi \in \Phi$. During the processing of mini batches, this algorithm adds some additional steps that are distinct from Algorithm 1. Prior to the main loop, fractional contributions to $\Delta^{(\tau)}$, for all τ, by the non-sensitive and sensitive attributes are determined. These contributions are represented by ψ_N and ψ_S, respectively, and are used in the computation of privacy budgets ϵ_N and ϵ_S. During the process of perturbation, noise is then added to each ∇J_ϕ in the update of ω_ϕ utilizing ϵ_N for $\phi \in \Phi_N$ and ϵ_S for $\phi \in \Phi_S$. We now prove that Algorithm 2 satisfies ϵ-differential privacy, as shown in Theorem 3.

Theorem 3. *Both Algorithm 2 (and therefore Algorithm 3) both satisfy ϵ-differential privacy.*

Proof. We compare the function f in the presence of B and B′ at some arbitrary point g.

$$
\begin{aligned}
\frac{\Pr\{f(\mathrm{B}_j)\}}{\Pr\{f(\mathrm{B}_j')\}} &= \prod_{k \in \Phi_N} \frac{\exp\left(-\frac{\epsilon_N}{\Delta^{(\tau)}} \left| f_k(\mathrm{B}_j) - g_k \right|\right)}{\exp\left(-\frac{\epsilon_N}{\Delta^{(\tau)}} \left| f_k(\mathrm{B}_j') - g_k \right|\right)} \\
&\quad \cdot \prod_{k \in \Phi_S} \frac{\exp\left(-\frac{\epsilon_S}{\Delta^{(\tau)}} \left| f_k(\mathrm{B}_j) - g_k \right|\right)}{\exp\left(-\frac{\epsilon_S}{\Delta^{(\tau)}} \left| f_k(\mathrm{B}_j') - g_k \right|\right)}
\end{aligned}
$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.3036422, IEEE Journal of Biomedical and Health Informatics

$$= \prod_{k\in\Phi_N} \exp\left(\frac{\epsilon_N\left(\left|f_k(\mathrm{B}_j') - g_k\right| - \left|f_k(\mathrm{B}_j) - g_k\right|\right)}{\Delta^{(\tau)}}\right)$$
$$\cdot \prod_{k\in\Phi_S} \exp\left(\frac{\epsilon_S\left(\left|f_k(\mathrm{B}_j') - g_k\right| - \left|f_k(\mathrm{B}_j) - g_k\right|\right)}{\Delta^{(\tau)}}\right)$$
$$\leq \prod_{k\in\Phi_N} \exp\left(\frac{\epsilon_N\left|f_k(\mathrm{B}_j) - f_k(\mathrm{B}_j')\right|}{\Delta^{(\tau)}}\right)$$
$$\cdot \prod_{k\in\Phi_S} \exp\left(\frac{\epsilon_S\left|f_k(\mathrm{B}_j) - f_k(\mathrm{B}_j')\right|}{\Delta^{(\tau)}}\right)$$
$$= \exp\left(\frac{\epsilon_N}{\Delta^{(\tau)}} \sum_{k\in\Phi_N} \left|f_k(\mathrm{B}_j) - f_k(\mathrm{B}_j')\right| + \frac{\epsilon_S}{\Delta^{(\tau)}} \sum_{k\in\Phi_S} \left|f_k(\mathrm{B}_j) - f_k(\mathrm{B}_j')\right|\right)$$
$$= \exp\left(\frac{\eta^{(\tau)}\epsilon_N}{b\Delta^{(\tau)}} \sum_{k\in\Phi_N} \left|\nabla\ell_k(\omega^{(\tau)}, t_b) - \nabla\ell_k(\omega^{(\tau)}, t_{b'})\right| + \frac{\eta^{(\tau)}\epsilon_S}{b\Delta^{(\tau)}} \sum_{k\in\Phi_S} \left|\nabla\ell_k(\omega^{(\tau)}, t_b) - \nabla\ell_k(\omega^{(\tau)}, t_{b'})\right|\right)$$
$$\leq \exp(\epsilon_N\psi_N + \epsilon_S\psi_S)$$
$$= \exp\left(\frac{\psi_N}{\psi_N + \gamma\psi_S}\epsilon + \frac{\gamma\psi_S}{\psi_N + \gamma\psi_S}\epsilon\right)$$
$$= \exp(\epsilon)$$

The first inequality follows from the triangle inequality. The second inequality follows from the application of Lemma 2 and Corollary 2, which defines ψ_N and ψ_S. Note that without loss of generality, the function f may receive D or D' as input. ■

Lemma 2. *Parameters ψ_N and ψ_S are at most $\max_t \frac{\|\nabla\ell_N\|_1}{\|\nabla\ell\|_1}$ and $\max_t \frac{\|\nabla\ell_S\|_1}{\|\nabla\ell\|_1}$, respectively.*

Proof.

$$\psi_N = \frac{\eta^{(\tau)}}{b\Delta^{(\tau)}} \sum_{k\in\Phi_N} \left|\nabla\ell_k(\omega^{(\tau)}, t_b) - \nabla\ell_k(\omega^{(\tau)}, t_{b'})\right| \leq \frac{2\eta^{(\tau)}}{b\Delta^{(\tau)}} \max_t \sum_{k\in\Phi_N} \left|\nabla\ell_k(\omega^{(\tau)}, t)\right|$$
$$= \max_t \frac{\sum_{k\in\Phi_N}\left|\nabla\ell_k(\omega^{(\tau)}, t)\right|}{\|\nabla\ell(\omega^{(\tau)}, t)\|_1} = \max_t \frac{\left\|\nabla\ell_N(\omega^{(\tau)}, t)\right\|_1}{\|\nabla\ell(\omega^{(\tau)}, t)\|_1}$$

By a similar set of steps, we also have the following,

$$\psi_S = \max_t \frac{\sum_{k\in\Phi_S}\left|\nabla\ell_k(\omega^{(\tau)}, t)\right|}{\|\nabla\ell(\omega^{(\tau)}, t)\|_1} = \max_t \frac{\left\|\nabla\ell_S(\omega^{(\tau)}, t)\right\|_1}{\|\nabla\ell(\omega^{(\tau)}, t)\|_1}$$

Note that both ψ_S and ψ_n are fractional contributions portions to the L^1 norm of the loss gradient for sensitive and non-sensitive attributes, respectively ($\psi_N + \psi_S = 1$). ■

Algorithm 2 Mosaic Gradient Perturbation

Input: Data D with labels Φ, parameters $\epsilon, \gamma, \psi_s, \Lambda, \mathrm{K}, b$
Output: Approximate noisy minimizer $\bar{\omega}$

Partition Φ into Φ_N and Φ_S
Initialize $\omega^{(1)}$, $\tau = 1$, $\kappa = 1$, $\eta_0 = \sqrt{\frac{1}{\Lambda^{1/2}}}$, $\psi_N = 1 - \psi_S$
Set $\epsilon_N = \frac{1}{\psi_N + \gamma\psi_S}\epsilon$, $\epsilon_S = \frac{\gamma}{\psi_N + \gamma\psi_S}\epsilon$
Set $\tau_0 = \frac{1}{\Lambda\eta_0}$
Distribute D into a set of batches B, each of size b
while $\kappa \leq \mathrm{K}$
  for each $j = 1, \ldots, |\mathrm{B}|$ **do**
    Set $\eta^{(\tau)} = \frac{1}{\Lambda(\tau_0 + \tau - 1)}$
    Set $\Delta^{(\tau)} = \frac{2\theta\eta^{(\tau)}}{b}$
    for each $\phi \in \Phi$ **do**
      if $\phi \in \Phi_N$ **then**
        Draw a constant $z_\phi^{(\tau)} \sim Lap\left(\frac{\Delta^{(\tau)}}{\epsilon_N}\right)$
      else
        Draw a constant $z_\phi^{(\tau)} \sim Lap\left(\frac{\Delta^{(\tau)}}{\epsilon_S}\right)$
      end if
      Set $\omega_\phi^{(\tau+1)} = \omega_\phi^{(\tau)} - \eta^{(\tau)}\left(\nabla J_\phi(\omega^{(\tau)}, \mathrm{B}_j) + \frac{1}{b}z_\phi^{(\tau)}\right)$
    end for
    Set $\tau = \tau + 1$
  end for
  Set $\kappa = \kappa + 1$
end while
Let $\bar{\omega} = \omega^{(\tau)}$

Corollary 2. *If $\|\nabla\ell\|_1 \leq \theta$, ψ_N and ψ_S can be freely configured arbitrarily, according to the constraint $\psi_N + \psi_S = 1$. Under some arbitrary t that maximizes $\|\nabla\ell\|_1$, let parameters ζ_N and ζ_S be $\|\nabla\ell_N\|_1$ and $\|\nabla\ell_S\|_1$, respectively.*

Proof.

$$\psi_N + \psi_S = 1$$
$$\max_t \frac{\left\|\nabla\ell_N(\omega^{(\tau)}, t)\right\|_1 + \left\|\nabla\ell_S(\omega^{(\tau)}, t)\right\|_1}{\|\nabla\ell(\omega^{(\tau)}, t)\|_1} = 1$$
$$\frac{\zeta_N + \zeta_S}{\theta} = 1$$
$$\zeta_N + \zeta_S = \theta$$

Since the choice of t is arbitrary, there are a multitude of choices that result in a maximized loss gradient. The choice of t will have implications on the value of ζ_N and ζ_S. The overall loss gradient $\|\nabla\ell\|_1$ may have different magnitudes of contribution from non-sensitive and sensitive attributes, $\zeta_N \leq \theta$ and $\zeta_S \leq \theta$. This fact combined with the result above shows that since the choice of t is arbitrary, the choice of ζ_N and ζ_S are also equally as arbitrary. ■

Algorithm 2 can be further extended into a distributed version, as shown in Algorithm 3. In this algorithm, iterations and epochs are synonymous as the entire dataset is utilized to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.3036422, IEEE Journal of Biomedical and Health Informatics

compute the gradient at each step. Prior to the calculation of the global sensitivity, the data are distributed among many computing units. Each computing unit j returns a loss gradient sum. These loss sums are then utilized to compute the gradient of the objective function. The algorithm then proceeds as Algorithm 2 would normally. The proof of Theorem 3 shows that this algorithm also maintains ϵ-differential privacy.

Algorithm 3 Distributed Mosaic Gradient Perturbation

Input: Data D with labels Φ, parameters $\epsilon, \gamma, \psi_S, \Lambda, \mathrm{K}, b$
Output: Approximate noisy minimizer $\bar{\omega}$

Partition Φ into Φ_N and Φ_S
Initialize $\omega^{(1)}$, $\tau = 1$, $\eta_0 = \sqrt{\frac{1}{\Lambda^{1/2}}}$, $\psi_N = 1 - \psi_S$
Set $\epsilon_N = \frac{1}{\psi_N + \gamma\psi_S}\epsilon$, $\epsilon_S = \frac{\gamma}{\psi_N + \gamma\psi_S}\epsilon$
Set $\tau_0 = \frac{1}{\Lambda\eta_0}$
Distribute D into a set of batches B, each of size b
while $\tau \leq \mathrm{K}$
  Transmit each batch $\mathrm{B}_j \subset \mathrm{B}$ to independent processors
  for all $j = 1, \ldots, |\mathrm{B}|$ **do**
    $V\left(\omega^{(\tau)}, \mathrm{B}_j\right) = \sum_{t_i \in \mathrm{B}_j} \nabla\ell(\omega^{(\tau)}, t_i)$
  end for
  $\nabla J\left(\omega^{(\tau)}, D\right) = \Lambda\nabla R(\omega) + \frac{1}{b|\mathrm{B}|}\sum_{j=1}^{|\mathrm{B}|} V\left(\omega^{(\tau)}, \mathrm{B}_j\right)$
  Set $\eta^{(\tau)} = \frac{1}{\Lambda(\tau_0 + \tau - 1)}$
  Set $\Delta^{(\tau)} = \frac{2\theta\eta^{(\tau)}}{b|\mathrm{B}|}$
  for each $\phi \in \Phi$ **do**
    if $\phi \in \Phi_N$ **then**
      Draw a constant $z_\phi^{(\tau)} \sim Lap\left(\frac{\Delta^{(\tau)}}{\epsilon_N}\right)$
    else
      Draw a constant $z_\phi^{(\tau)} \sim Lap\left(\frac{\Delta^{(\tau)}}{\epsilon_S}\right)$
    end if
    Set $\omega_\phi^{(\tau+1)} = \omega_\phi^{(\tau)} - \eta^{(\tau)}\left(\nabla J_\phi\left(\omega^{(\tau)}, D\right) + \frac{1}{b|\mathrm{B}|} z_\phi^{(\tau)}\right)$
  end for
  Set $\tau = \tau + 1$
end while
Let $\bar{\omega} = \omega^{(\tau)}$

IV. Experimental Design and Materials

4.1 Design of Experiments

In this study, the proposed MGP methodology is evaluated and validated with a real-world ICU dataset that consist of 4,000 patient records from 48 hours of ICU stays. This dataset is extracted from the Multiparameter Intelligent Monitoring in Intensive Care (MIMIC) II Clinical Database [20], which was developed to advance intelligent patient monitoring research in the critical care environment. These algorithms are also compared against both standard Output Perturbation (OutP) and Objective Perturbation (ObjP).

There are multiple preprocessing steps taken before the implementation and testing of private algorithms. First, the data are transformed through feature extraction and selection per Chen *et al.* [12]. Second, data columns are normalized such that the domain of the input variables is between negative one and one. Next, each tuple's input space is confined to the L^1 norm, $\|\mathbf{x}_i\|_1 \leq 1$. The data are then bootstrapped to balance the observed outcomes. Then, performance metrics (i.e., accuracy, sensitivity, and specificity levels) are acquired from a non-private model, which is then used as a baseline to estimate the risk of patient mortality. Further, Gaussian noise is added into the data to simulate different levels of uncertainty in realistic circumstances.

Next, a privacy-preserving algorithm is implemented and tested. The model accuracy of the algorithm is calculated by dividing the number of cases that are correctly classified by the total number of data tuples. Attack accuracy is determined by calculating the coefficient of determination, $R^2 = 1 - \frac{\sum_{\alpha\in D}(x_{\alpha s} - \hat{x}_{\alpha s})^2}{\sum_{\alpha\in D}(x_{\alpha s} - \bar{x}_{\alpha s})^2}$, where $\hat{x}_{\alpha s}$ is an estimate of the sensitive attribute's value and $\bar{x}_{\alpha s}$ is the sample average value of the sensitive attribute.

The gains in computational efficiency when utilizing the distributed version of the MGP algorithm are also evaluated and benchmarked in this investigation. The distributed algorithm is compared against its serial-computing counterpart for different sizes of patient datasets.

4.2 Configuration

In the experimentation, the total number of epochs per scenario was set to $\mathrm{K} = 1000$. Batch sizes were set to $b = 500$ and the regularization parameter was initialized to be $\Lambda = 0.0001$. In the case of the MGP method, the value of ψ_S was configured to be proportional to the dimensionality of sensitive attributes, which in this case is $1/45$. Each scenario was replicated 500 times. Model inversion was then performed on the full dataset utilizing the selected model. Each experimental case was then benchmarked against the performance of the baseline model, where no perturbation is applied to the gradient. Standard GP is conducted first, where ϵ is varied between 10^0 and 10^{-5}. Next, MGP is carried forth by varying γ between 10^0 and 10^{-7}, with $\epsilon = 1$.

For the evaluation of the distributed algorithm's computational efficiency, the number of patients included in the dataset is varied between 6,892 and 82,704. The value of K, b, Λ, and ψ_S are kept the same as the previous set of experiments. The value of ϵ and γ are kept constant at 1 and 0.5, respectively. Experiments are run in Python 3.6 on a Red Hat Linux machine with 128 GB of RAM and an Intel Xeon E5-2650v4 processor running at 2.2 GHz. Serial processing constrains the machine to utilizing a single core. Distributed processing is carried forth by sharing the workload among multiple cores. In the case of this investigation, distributed processing scenarios are run with 8 and 16 cores at a time.

V. Experimental Results

Figure 4 shows the impact of varying ϵ on the model and attack accuracy. Decreasing ϵ causes both the model and attack accuracies to decrease. However, the attack accuracy decreases at a much more substantial rate once ϵ drops beneath 10^{-2}. As ϵ approaches 10^{-4}, attack accuracy approaches zero with a corresponding model accuracy decrease of around 5% from the baseline value.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.3036422, IEEE Journal of Biomedical and Health Informatics

The decay in model accuracy seems to accelerate as ϵ decreases beneath 10^{-4}, though it should be noted that decreasing ϵ any further has no additional value since the attack accuracy has already been reduced to zero. Overall, this result replicates what has been previously seen for the standard GP method. In most use cases, this decrease in accuracy may be considered insignificant. However, given that the application domain in the ICU, it is desirable to maintain model performance while mitigating the privacy risk due to the implications on mortality outcomes.

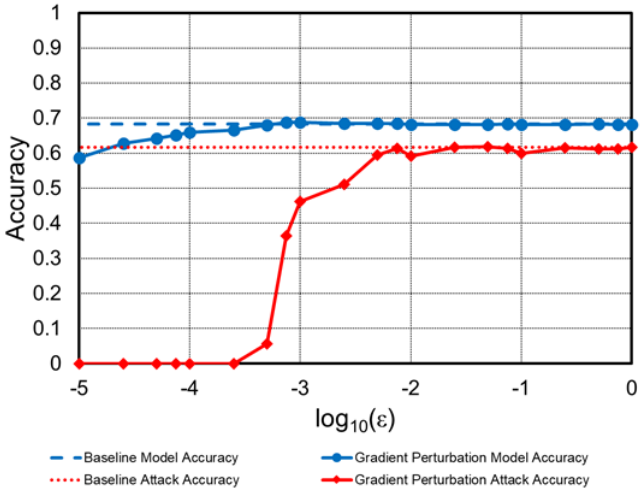

Figure 4. Model and attack accuracy of GP under varying ϵ.

The results for OutP are shown in Figure 5. It may be noted that almost all experimental cases for OutP under differential privacy show a model accuracy of ~50% and an attack accuracy of ~0%. By contrast, Figure 6 demonstrates that ObjP yields a better utility with the model accuracy starting to significantly decay when ϵ decreases below 10^{-2}. The attack accuracy does not start to decay until ϵ drops below 10^{-3}. Similar to GP, the attack accuracy approaches zero as ϵ approaches 10^{-4}. As this occurs, however, ObjP's model accuracy experiences a decrease of ~10%.

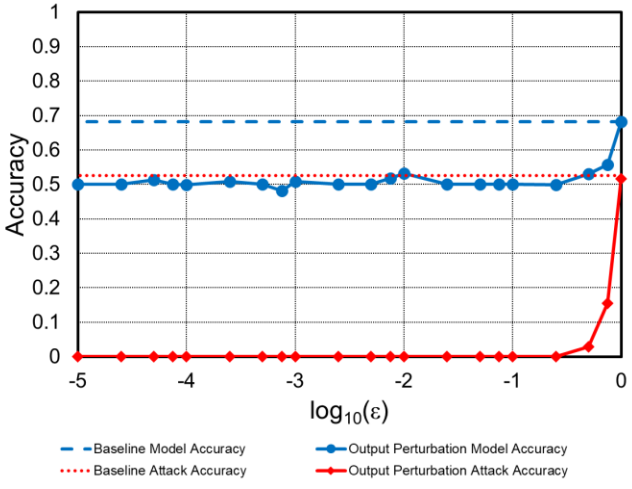

Figure 5. Model and attack accuracy of OutP under varying ϵ.

As a result, ObjP has a higher cost to model utility when mitigating the risk of model inversion when compared to GP. It may also be noted that OutP and ObjP methods tend to be limited in their ability to handle distributed processing. There is also a unique advantage to develop new privacy algorithms with gradient-based methods, which are more suitable for distributed computing.

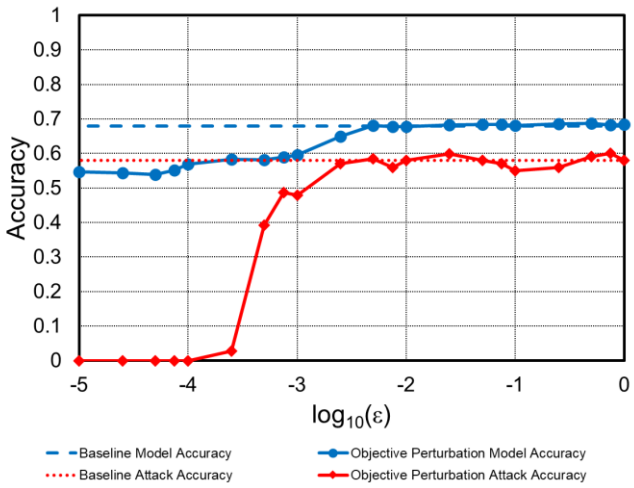

Figure 6. Model and attack accuracy of ObjP under varying ϵ.

Varying γ for the MGP algorithm will produce different values for ϵ_N and ϵ_S. The results shown in Table II displays the resultant impact of decreasing the order of magnitude of γ while ϵ is kept constant. It may be noted that the value of ϵ_N is largely unchanged. By contrast, ϵ_S is shown to have an order of magnitude that is consistent with that of γ.

TABLE II. THE VARIATIONS OF PRIVACY BUDGETS ϵ_N AND ϵ_S WITH RESPECT TO γ FOR $\epsilon = 1$

γ	ϵ_N	ϵ_S
10^{0}	1.000	$1.000 \cdot 10^{0}$
10^{-1}	1.020	$1.020 \cdot 10^{-1}$
10^{-2}	1.022	$1.022 \cdot 10^{-2}$
10^{-3}	1.023	$1.023 \cdot 10^{-3}$
10^{-4}	1.023	$1.023 \cdot 10^{-4}$
10^{-5}	1.023	$1.023 \cdot 10^{-5}$
10^{-6}	1.023	$1.023 \cdot 10^{-6}$
10^{-7}	1.023	$1.023 \cdot 10^{-7}$

These values of ϵ_N and ϵ_S are utilized to produce the results seen in Figure 7. As one can see, both the attack and model accuracy decrease as γ decreases. Nonetheless, the decay in attack accuracy is substantially greater. As γ decreases beneath 10^{-4}, attack accuracy begins an observable descent. Once γ decreases below 10^{-6}, the attack accuracy approaches zero. Note that for all experimental cases, the model accuracy does not decrease more than 1% from the baseline.

Thus, the ramifications of varying γ are less severe than those associated with varying ϵ. In both experimental cases, it is possible to reduce the attack accuracy to zero. Therefore, if seeking to mitigate model inversion risk, decreasing γ is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.3036422, IEEE Journal of Biomedical and Health Informatics

more preferred. Nonetheless, it should be noted that an acceptable level of ϵ still needs to be established in order to uphold the privacy of dataset inclusion.

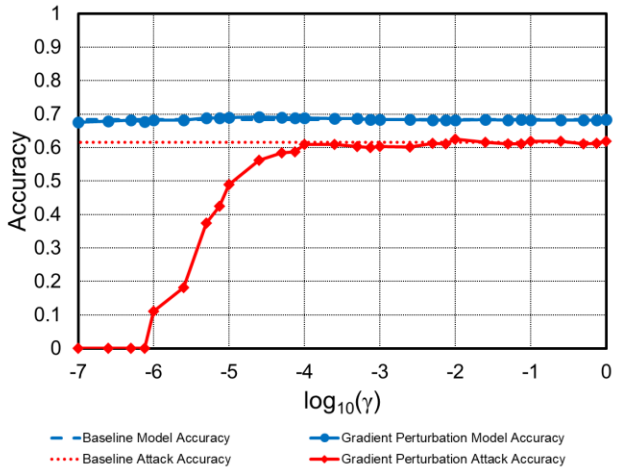

Figure 7. Model and attack accuracy of MGP under varying γ ($\epsilon = 1$).

Figure 8 shows the evaluation and comparison of computational efficiency between distributed and serial-computing versions of the MGP algorithm. As the number of patients increases, the gap in performance between serial-computing and distributed cases (i.e., either 8 or 16 cores) widens substantially. Nonetheless, even when the number of patients is closer to 6,892, utilizing any form of distributed processing is significantly faster than serial computation. When the dataset's size increases to 82,704, both distributed computing scenarios are significantly more efficient than the serial version. Note that the 16 core scenarios are about twice as fast as the 8 core scenarios.

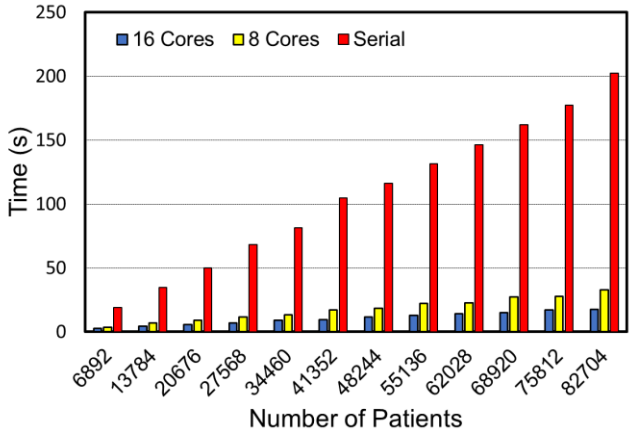

Figure 8. Performance comparison of computational efficiency between distributed and serial-computing versions of MGP algorithms.

VI. Discussion

MGP introduces an additional tuning parameter, γ, which provides superior performance in the risk mitigation of model inversion attacks when compared with ϵ adjustments in existing differential privacy techniques. Despite the promise of MGP, there are a few potential limitations, including the necessity of $\|\nabla\ell\|_1$ to be bounded by θ. It may be necessary to project any $\nabla\ell$ such that the assumption is upheld; certain loss functions like the quadratic loss function have unbounded gradients. Also, the distributed version of MGP algorithm provides privacy perturbation after loss sums are aggregated. Future work will focus on the investigation of distributed perturbation that can be designed to offer addition levels of privacy protection, especially as the intricacy of cyber-attacks increases over time.

An attractive feature of the MGP algorithm is differential privacy protection in a distributed setting. This is targeted at IoT devices and telemedicine systems that are increasingly deployed in the healthcare setting. A pandemic, e.g., COVID-19, further drives the widespread applications of telemedicine systems with IoT sensors and devices. The new paradigm leads to the proliferation of large amounts of data in a distributed manner. Traditional practices focus more on the effectiveness of data analytics (e.g., accuracy), but are less concerned about data privacy and model inversion attacks. Privacy algorithms that can further provide distributed capabilities will be more important as there are bottlenecks associated with serial processing. These bottlenecks can be seen in the case study section, specifically in Figure 8.

In addition, MGP has strong potentials to be used as a supporting technology in a Consumer-Owned Data Marketplace (CODM). The adverse effects of data breaches have resulted in public concerns, new government regulations, and the expansion of business responsibilities. Given this looming shadow over the existing data economy, privacy-preserving CODM can become a viable alternative to help create a data exchange that empowers both consumers and businesses alike. The underlying philosophy behind CODM is that people own their own data and should control its usage. Distributed differential privacy techniques such as MGP can provide strong support for this endeavor.

VII. Conclusions

Contemporary advances in the IoT have enabled the development of a smart health system, powered by the internet-like connection of healthcare professionals, patients, medical devices, insurance companies, and hospitals. Privacy concerns arise from the increasing ubiquity of data transmission due to these innovations as well as the ever-advancing complexity of cyber-attacks. In the present investigation, we develop the MGP technique as a means to uphold differential privacy and reduce the risk of model inversion attacks. The gradient-based nature of the MGP technique can be naturally extended to accommodate for distributed processing of larger datasets. When doing so, the algorithm is more computationally efficient.

The MGP technique introduces a new control parameter, γ, that allows one to control the ratio of noise between non-sensitive and sensitive attributes. Due to the inclusion of this γ parameter, the technique is capable of reducing the accuracy of inversion attacks to zero while impacting model accuracy less significantly than standard GP approaches. Within the domain of the ICU, it is desirable to maintain model performance while mitigating privacy risks due to the implications of mortality outcomes. The results delivered by

this investigation gives credence to the assertion that the MGP technique can maintain several views of patient privacy with minimal tradeoffs to patient care.

REFERENCES

[1] C. Kan, Y. Chen, F. Leonelli, and H. Yang, "Mobile sensing and network analytics for realizing smart automated systems towards health Internet of Things," in *2015 IEEE International Conference on Automation Science and Engineering (CASE)*, Aug. 2015, pp. 1072–1077, doi: 10.1109/CoASE.2015.7294241.

[2] Statista, "Projected size of the global market for wearable devices in the healthcare sector from 2015 to 2021 (in millions of U.S. dollars)," 2019. [Online]. Available: https://www.statista.com/statistics/607982/healthcare-wearable-device-revenue-worldwide-projection/.

[3] H. Yang, C. Kan, A. Krall, and D. Finke, "Network Modeling and Internet of Things for Smart and Connected Health Systems - A Case Study for Smart Health Monitoring and Management," *IISE Transaction on Health Systems Engineering*, pp. 1–28, 2019.

[4] L. Liu, J. Xu, Y. Huan, Z. Zou, S. Yeh, and L. Zheng, "A Smart Dental Health-IoT Platform Based on Intelligent Hardware, Deep Learning, and Mobile Terminal," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 3, pp. 898–906, Mar. 2020, doi: 10.1109/JBHI.2019.2919916.

[5] H. Zhou, H. Lv, Z. Pang, X. Huang, H. Yang, and G. Yang, "IoT-enabled Dual-arm Motion Capture and Mapping for Telerobotics in Home Care," *IEEE Journal of Biomedical and Health Informatics*, pp. 1–1, 2019, doi: 10.1109/JBHI.2019.2953885.

[6] N. Cai and S. Kou, "Econometrics with Privacy Preservation," *Operations Research*, vol. 67, no. 4, pp. 905–926, 2019, doi: 10.1287/opre.2018.1834.

[7] S. Walker-Roberts, M. Hammoudeh, and A. Dehghantanha, "A Systematic Review of the Availability and Efficacy of Countermeasures to Internal Threats in Healthcare Critical Infrastructure," *IEEE Access*, vol. 6, pp. 25167–25177, 2018, doi: 10.1109/ACCESS.2018.2817560.

[8] C. Dwork and A. Roth, "The Algorithmic Foundations of Differential Privacy," *Found. Trends Theor. Comput. Sci.*, vol. 9, no. 3–4, pp. 211–407, Aug. 2014, doi: 10.1561/0400000042.

[9] M. Fredrikson, E. Lantz, S. Jha, S. Lin, D. Page, and T. Ristenpart, "Privacy in Pharmacogenetics: An End-to-End Case Study of Personalized Warfarin Dosing," *Proc USENIX Secur Symp*, vol. 2014, pp. 17–32, Aug. 2014.

[10] L. Sweeney, *Matching Known Patients to Health Records in Washington State Data*. 2013.

[11] L. Sweeney, A. Abu, and J. Winn, *Identifying Participants in the Personal Genome Project by Name (A Re-identification Experiment)*. 2013.

[12] Y. Chen, F. Leonelli, and H. Yang, "Heterogeneous Sensing and Predictive Modeling of Postoperative Outcomes," in *Healthcare Analytics*, John Wiley & Sons, Ltd, 2016, pp. 463–501.

[13] K. Chaudhuri and C. Monteleoni, "Privacy-preserving logistic regression," 2008.

[14] J. Zhang, Z. Zhang, X. Xiao, Y. Yang, and M. Winslett, "Functional Mechanism: Regression Analysis under Differential Privacy," presented at the VLDB Endowment, 2012.

[15] Y. Wang, C. Si, and X. Wu, "Regression Model Fitting under Differential Privacy and Model Inversion Attack," presented at the International Joint Conference on Artificial Intelligence, Buenos Aires, 2015.

[16] N. Li, M. Lyu, D. Su, and W. Yang, "Differential Privacy: From Theory to Practice," *Synthesis Lectures on Information Security, Privacy, and Trust*, vol. 8, pp. 1–138, 2016, doi: 10.2200/S00735ED1V01Y201609SPT018.

[17] J. L. Raisaro *et al.*, "Protecting Privacy and Security of Genomic Data in i2b2 with Homomorphic Encryption and Differential Privacy," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 15, no. 5, pp. 1413–1426, Sep. 2018, doi: 10.1109/TCBB.2018.2854782.

[18] N. Almadhoun, E. Ayday, and Ö. Ulusoy, "Differential privacy under dependent tuples—the case of genomic privacy," *Bioinformatics*, vol. 36, no. 6, pp. 1696–1703, Mar. 2020, doi: 10.1093/bioinformatics/btz837.

[19] S. Song, K. Chaudhuri, and A. D. Sarwate, "Stochastic gradient descent with differentially private updates," in *2013 IEEE Global Conference on Signal and Information Processing*, Dec. 2013, pp. 245–248, doi: 10.1109/GlobalSIP.2013.6736861.

[20] M. Saeed *et al.*, "Multiparameter Intelligent Monitoring in Intensive Care II: a public-access intensive care unit database," *Crit Care Med*, vol. 39, no. 5, pp. 952–960, May 2011, doi: 10.1097/CCM.0b013e31820a92c6.