Enhancing Privacy in Machine Learning: A Robust Approach for Preventing Attribute Inference Attacks

Myria Bouhaddi and Kamel Adi

Computer Security Research Laboratory, University of Quebec in Outaouais, Gatineau, Quebec, Canada

Keywords: Machine Learning Security, Attribute Inference Attacks, Confidence Masking, Adversarial Machine Learning.

Abstract: Machine learning (ML) models, widely used in sectors like healthcare, finance, and smart city development, face significant privacy risks due to their use of crowdsourced data containing sensitive information. These models are particularly susceptible to attribute inference attacks, where adversaries use model predictions and public or acquired metadata to uncover sensitive attributes such as locations or political affiliations. In response, our study proposes a novel, two-phased defense mechanism designed to efficiently balance data utility with privacy. Initially, our approach identifies the minimal level of noise needed in the prediction score to thwart an adversary's classifier. This threshold is determined using adversarial ML techniques. We then enhance privacy by injecting noise based on a probability distribution derived from a constrained convex optimization problem. To validate the effectiveness of our privacy mechanism, we conducted extensive experiments using real-world datasets. Our results indicate that our defense model significantly outperforms existing methods, and additionally demonstrates its adaptability to various data types.

1 INTRODUCTION

The rapid revolution of artificial intelligence, particularly in deep learning, has marked a significant shift across various sectors including computer vision, healthcare, autonomous driving, and natural language processing. In recent times, prominent technology companies such as Google, Microsoft, Amazon, and IBM have made these models available through APIs. This development means that a broad audience can now access and implement advanced AI without the necessity of developing models from the ground up. This democratization of AI has ignited unparalleled innovation, especially since many ML technologies are based on proprietary datasets that span domains such as personalized medicine (Weiss et al., 2012), product recommendation (Linden et al., 2003), finance (Dunis et al., 2016), law (Hildebrandt, 2018) and social networks (Farnadi et al., 2016).

However, as the deployment of these models grows, there is an increased need for extensive datasets to train them effectively. This rise in data requirements intensifies privacy concerns, complicating their implementation (Hu et al., 2022; Bouhaddi and Adi, 2023). Such concerns are further accentuated by risks associated with data privacy, most notably through inference attacks that compromise the confidentiality of training data, revealing sensitive information via methods like membership, attribute (Shokri et al., 2017), property inference (Ateniese et al., 2015), and partial memorization (Carlini et al., 2019). Particularly, Attribute Inference Attacks (AIA) exploit partial knowledge of training records to deduce sensitive attributes from model predictions, a technique akin to data imputation but uniquely leveraging the model's learned patterns in addition to the available data.

Recent studies, such as those by Fredrikson *et al.* (Fredrikson et al., 2014), have further categorized them into two main categories: Model Inversion Attribute Inference (MIAI) attacks and Typical Instance Reconstruction (TIR) attacks. While the former focuses on discerning sensitive attributes of individuals used in training, the latter attempts to generate a 'typical' instance for a specific class, e.g., reconstructing a facial image similar to a target individual.

Illustratively, consider healthcare care, a domain rich in sensitive information. ML models that predict patient outcomes or aid diagnostics may use personal health data. Exploiting a model inversion attack, adversaries can infer patients' health histories from predictions, even without names. This breach compromises confidentiality, allowing, hence, misuse like discriminatory insurance practices.

Bouhaddi, M. and Adi, K.
Enhancing Privacy in Machine Learning: A Robust Approach for Preventing Attribute Inference Attacks.
DOI: 10.5220/0012768700003767

In *Proceedings of the 21st International Conference on Security and Cryptography (SECRYPT 2024)*, pages 224-236
ISBN: 978-989-758-709-2; ISSN: 2184-7711

In this paper, we specifically examine attribute inference vulnerabilities in classification models used in Machine Learning as a Service (MLaaS), where private data about individuals are employed. We focus on scenarios where an adversary, with black-box access to an MLaaS model, attempts to infer sensitive attributes of a target individual. By using prediction scores from these machine learning models, an adversary can apply a classifier to accurately predict a user's gender, demonstrating the technique's surprising precision (Weinsberg et al., 2012). The success of attribute inference attacks in identifying sensitive attributes from publicly accessible data hinges on the statistical correlations between a user's private and public information.

In this context, the attackers do not have direct visibility into the model's internal workings or its training algorithms. Instead, they have access to the model's prediction scores and, potentially, partial information regarding the training dataset and its probability distribution. This accessibility allows for a nuanced understanding of the model's output behavior in response to various inputs, providing a covert pathway for sophisticated attacks. Such a scenario posits significant challenges in safeguarding sensitive information, as it exposes the model to indirect inference attacks, where adversaries can cleverly deduce sensitive attributes from seemingly innocuous prediction scores.

In addressing the pressing issue of privacy within the ML as a service framework, it becomes evident that traditional privacy-preserving methods, such as differential privacy, fall short when it comes to mitigating the risk of attribute inference attacks (Jayaraman and Evans, 2019; Jayaraman and Evans, 2022). Although differential privacy provides a mathematical guarantee against the identification of individuals within a dataset, it struggles to maintain a beneficial balance between data utility and privacy, often resulting in reduced model accuracy. Furthermore, the complexity of managing noise in prediction scores to maintain differential privacy complicates its application in practical settings (Dwork et al., 2014). This method does not directly counter the nuanced threat posed by adversaries leveraging prediction scores to infer sensitive attributes, thereby highlighting the limitations of conventional approaches in the face of evolving attack vectors within machine learning applications.

Exploring alternative solutions, game-theoretic approaches emerge as a promising avenue for devising strategic defenses against inference attacks (Shokri, 2014; du Pin Calmon and Fawaz, 2012). By modeling the interactions between attackers and defenders as a game, these methods aim to predict and neutralize adversarial strategies. However, despite their potential for creating dynamic and adaptive defense mechanisms, the implementation of game-theoretic solutions in real-time applications presents significant challenges. The complexity and computational demands of these approaches often render them impractical for deployment in scenarios requiring immediate response, underscoring the need for more efficient and scalable solutions.

In this context, defense techniques centered on score masking strategies offer a viable and effective means of preserving privacy (Jia and Gong, 2018). By manipulating or obscuring the prediction scores provided by ML models, these techniques can significantly reduce the risk of sensitive attribute inference without compromising the utility of the model's output. One significant advantage of score masking is that it can be applied to existing classifiers without the need for retraining, which avoids the often costly and time-consuming process of updating models. The adaptability of score masking methods allows for tailored applications across various contexts and constraints, ensuring that privacy measures can be implemented in a manner that aligns with the specific requirements and risks of each scenario. This flexibility, coupled with the relative ease of implementation, positions score masking as a cornerstone of privacy preservation in the ML as a service paradigm.

In the present work, we propose a practical solution designed to operate efficiently in real-time applications, effectively perturbing the prediction score vector of the target ML model. This perturbation aims to randomize the adversary's sensitive attribute classifier's predictions, constituting the first phase of our solution. We propose leveraging adversarial machine learning, traditionally viewed as an offensive technique, as a defensive tool. Specifically, we use adversarial examples to deceive the adversary, a novel application that turns the tables on traditional attack vectors. The primary challenge of this approach is to maintain the predictive utility of the score vector, ensuring it continues to accurately predict the correct class while simultaneously safeguarding privacy. To address the absence of the adversary's sensitive attribute classifier, our defense constructs its classifier to link the score vector with the sensitive attribute value. Given the shared classification boundaries between the adversary's and our classifiers, the principle of transferability in adversarial machine learning (Papernot et al., 2017; Liu et al., 2016) ensures that a score vector perturbed for our defense classifier is also effective against the adversary's classifier. The second phase involves devising a mechanism to determine the optimal probability of a noise vector, constrained by the utility of the perturbed score vector, essentially framing it as an optimization problem to be solved.

In summary, our contributions are as follows:

- We propose a practical model designed to mask the prediction score vector of the target ML model. This approach aims to randomize the sensitive attribute predictions by an adversary's classifier, leveraging adversarial machine learning techniques—typically seen as offensive tools—in a defensive capacity. Our method focuses on manipulating the score vector in such a way that it confounds the attacker's efforts while preserving the utility of the predictions for legitimate uses.
- Our algorithm (called NOISY) leverages the Jacobian based Saliency Map Attack (JSMA) technique to craft adversarial examples that disrupt adversaries' attribute inference efforts. By integrating JSMA's approach to inject noise into the prediction score vector, NOISY renders attempts to infer sensitive attributes as unreliable as random guesses, all while preserving the original model's accuracy.
- Through rigorous experimentation, we have substantiated the efficacy of our approach, witnessing its prowess across diverse datasets, thereby reinforcing its practicality and applicability in real-world scenarios.

2 RELATED WORK

In this section, we provide an exploration of the landscape surrounding sensitive attribute inference attacks. We define these attacks, their different strategies, and the methods used to address them.

2.1 Attribute Inference Attacks

Recent studies (Kosinski et al., 2013; Gong and Liu, 2016; Weinsberg et al., 2012; Fredrikson et al., 2014) have revealed that users are at risk of attribute inference attacks, which aim to exploit machine learning models to expose sensitive information. Using publicly available data, attackers can deduce sensitive attributes of individuals, including but not limited to their gender, location, or political views. The core issue arises from the tendency of machine learning models to inadvertently leak sensitive information during the prediction process. Malicious entities are thus able to extract private or sensitive data from readily available sources, such as model predictions. This ability to infer details about the training data or inputs, which would otherwise remain hidden without the model's intervention, poses significant privacy challenges.

The sensitive attribute inference attacks can be broadly categorized into two main types: imputation-based and representation-based attacks. Both these classifications employ distinct strategies, assumptions, and techniques to target and exploit vulnerabilities inherent within machine learning models.

2.1.1 Imputation-Based Attacks

Focusing on the strategy of harnessing non-sensitive attributes, imputation-based attacks use the model's predictions and contextual data, such as the marginal prior over a sensitive attribute and the confusion matrix. The core objective of these attacks is to employ statistical inference in order to derive or impute concealed or missing data.

Jayaraman and Evans (Jayaraman and Evans, 2022) challenged the belief that standard blackbox imputation-based attacks outperformed others, showing they were equivalent to data imputation. Their research highlighted the distinction between authentic privacy risks and simple statistical deductions. Fredrikson *et al.* (Fredrikson et al., 2014) created a method based on target classifier responses to crafted inputs, evaluating the probability of correct confidential attribute values based on model feedback. Yeom *et al.* (Yeom et al., 2018) assumed a distribution over the confidential attribute, uncovering various attribute inference strategies, each with unique assumptions. Building on Fredrikson's work, Mehnaz *et al.* (Mehnaz et al., 2022) argued that model output precision is highest when matched with the right sensitive attribute in training. They proposed two attacks: the "Confidence only attack" that uses model confidence for sensitive attribute deduction, and the "Label-only attack" that zeroes in on select data entries to understand relationships between attributes.

2.1.2 Representation-Based Attacks

Representation-based attacks are particularly notable for their adeptness at leveraging the discernible disparities found within intermediate layer outputs or predictions, making them highly attuned to changes in attribute values. A clear illustration of this is seen in the distinct prediction output distributions associated with gender classifications, such as distinguishing between male and female.

Research by Song *et al.* (Song and Shmatikov, 2019) and Mahajan *et al.* (Mahajan et al., 2020) are based on the premise that the training data employed by machine learning models do not explicitly contain sensitive attributes. They design adversarial models to reverse-engineer the main model's outputs to reveal these attributes, typically using a 0.5 classification threshold. In contrast, Malekzadeh *et al.* (Malekzadeh et al., 2021) introduced a method using a custom loss function, aiming to embed the sensitive attribute in the model's output for easy retrieval during inference. This approach suggests a potential malicious intent by the model's creator, similar to implanting a system ”backdoor” to later reveal the sensitive attribute.

2.2 Mitigation Strategies for Sensitive Attribute Inference Attacks

Mitigation strategies for countering sensitive attribute inference attacks in machine learning (ML) models have become increasingly sophisticated as the threat landscape evolves.

Game-theoretic frameworks offer a strong theoretical foundation for privacy but are often computationally intensive, as noted by Shokri *et al.* (Shokri et al., 2016). For more practical applications, Salamatian *et al.*'s Quantization Probabilistic Mapping (QPM) simplifies the defense model for better efficiency (Salamatian et al., 2015).

Cryptography advancements like Homomorphic Encryption (HE) and Fully Homomorphic Encryption (FHE) allow computations on encrypted data, ensuring that cloud servers can process data without privacy breaches (Rivest et al., 1978; Chen et al., 2021). While these methods offer promising pathways to secure data processing, they are not without their challenges. Specifically, HE and FHE are known for their significant computational overhead, leading to increased processing time and energy consumption. This computational intensity can limit their practicality for real-time applications or those requiring rapid data processing, presenting a notable barrier to their widespread adoption in MLaaS contexts.

Differential Privacy (DP) offers a mathematically grounded method for protecting individual privacy by introducing noise into the dataset, thus masking the contributions of individual data points (Jayaraman and Evans, 2022). While DP is founded on solid mathematical principles ensuring privacy (Abadi et al., 2016), it tends to provide suboptimal solutions from a utility perspective (Jia et al., 2019). The added noise, though beneficial for privacy, can significantly reduce the utility of the data, making it less effective for certain analyses, especially where precision is critical. This trade-off highlights a fundamental challenge in privacy-preserving data analysis: balancing the need for robust privacy protection with the imperative to maintain data utility. Together, these approaches offer a multifaceted defense, tailored to balance privacy preservation with the practical demands of ML deployment.

3 PROBLEM FORMULATION

In our problem formulation, we clearly define the roles of three critical entities: the machine learning model, the attacker and the defense mechanism.

The *machine learning model* operates as the central system in our study. It is rigorously trained on user data with the sole aim of delivering precise and efficient predictions. This model, given its exposure and access to vast amounts of user data, inadvertently becomes a prime target for malicious entities.

The *attacker*, a malicious entity with a singular mission: to exploit the machine learning model. Its objective is clear: leverage the model's predictions to uncover private and potentially sensitive user attributes.

The *defense mechanism* emerges as the system's shield. It is meticulously designed to efficiently counter the attacker. This is achieved by ingeniously altering the score prediction. The catch, however, is to ensure that while the attacker is misled, the core utility and efficiency of the model remain untouched and unharmed.

3.1 Machine Learning Service Provider Model

A machine learning model is commonly viewed as a deterministic function:

$$f : X \to Y \tag{1}$$

The input of this function is a d-dimentional vector $x = [x_1, x_2, \cdots, x_d] \in X = \mathbb{R}^d$, representing d non-sensitive input attributes. For regression tasks, the output space Y is defined as the set of real numbers, $Y = \mathbb{R}$. However, our focus is on classification tasks, where the output space is distinct.

In the context of classification, the function $f : \mathbb{R}^d \to Y$ maps first the input vector x to a set of confidence scores $\upsilon = [\upsilon_1, \upsilon_2, \ldots, \upsilon_m]$, where each υ_j represents the model's confidence in assigning the j^{th} class label to x. The predicted class label, y, is then determined by selecting the label associated with the highest confidence score in υ, formally represented as $y = \arg\max_j \upsilon_j$, where $j \in \{1, \ldots, m\}$. Here, Y is the set of possible labels $\{y_1, y_2, \ldots, y_m\}$.

The model's parameters, denoted by θ, are iteratively refined based on the gradient of the loss function, which quantifies the discrepancy between the model's predictions $f(x;\theta)$ and the actual labels. . Training uses the dataset $\mathcal{D} \subseteq X \times Y$, aiming to optimize θ so that $f(x;\theta)$ can accurately map inputs x to their corresponding labels. Consequently, for any input x, the model's prediction is given as $f(x;\theta)$, where θ are the parameters refined through training to ideally minimize the loss function.

We consider p as a sensitive attribute belonging to the set P. An individual, linked to a data record within our training dataset, aims to keep this attribute p confidential. We suppose that this attribute p can assume k distinct values and our input attributes $x \in X$ are deemed non-sensitive. P represents the comprehensive set of all possible values that the sensitive attribute can take. Consequently, a data record is encapsulated as $z = (x, p, y)$, where x stands for the non-sensitive attributes, p represents the sensitive attribute, and y corresponds to the classification label. The association is defined as $(x, p) \in X \times P$.

While data is typically considered "public" for the purpose of training machine learning models, there exist scenarios in which "sensitive attributes" can be deduced from it. In one hand, these sensitive attributes might be used by the machine learning model to enhance prediction accuracy, raising the possibility that the model retains some memory of this information, which an adversary could exploit by looking for traces within the model's predictions. In another hand, the machine learning model may not have directly encountered or utilized this sensitive attribute; however, there could still be a link between this attribute and the public data of a record, which might allow for inference by an adversary. Data owners share their information for machine learning applications with the expectation that such sensitive details remain concealed, relying on the commitment to confidentiality of these attributes.

Let $\mathcal{D} \subseteq X \times Y$ be the training dataset of the target model, denoted f_{target}. In subsequent discussions, $y \in Y$ denotes the real value in $\mathcal{D}$, whereas $y' = f_{target}(x)$ corresponds to the model's prediction. A congruence between y and y' signifies accurate prediction by the model, whereas a discrepancy highlights a predictive error.

In the context of our study, once the model is trained on the dataset $\mathcal{D} = \{(x_i, y_i), i = 1..n\}$, it is deployed as an Application Programming Interface (API). Transitioning machine learning models into services via APIs is known as Machine Learning as a Service (MLaaS). MLaaS simplifies the use of these trained models, eliminating the intricacies of training and backend infrastructure. Yet, this convenience also amplifies data privacy concerns. MLaaS models are vulnerable to various inference attacks, underscoring the need for robust defense strategies.

3.2 Attacker

In our scenario, we introduce the presence of an adversarial entity $\mathfrak{A}$ intent on exploiting MLaaS facilities. This adversary interacts with the MLaaS by sending a series of queries and, in return, receives the associated confidence score vectors. Furthermore, this adversary has previously trained a classification model f_{adv} using supervised learning techniques. This prior training was facilitated by data they acquired, often from users who inadvertently disclosed their sensitive attributes.

The adversary is assumed to have access to a significant amount of information. Specifically, he owns all or a subset of the following capabilities/knowledge:

- Capability to interact with the target model, treated as a black-box. Specifically, the adversary can submit an inputs $x = [x_1, x_2, \ldots, x_d]$ to obtain the associated class label y'.
- Insight into the target model's confidence scores across m distinct class labels, υ.
- Knowledge of comprehensive or selective information about the non-sensitive attributes, while the sensitive aspect remains concealed.
- Availability of a supplementary dataset, $\mathcal{D}_{aux}$, originating from a similar data distribution as $\mathcal{D}$, on which f_{target} is trained. Notably, $\mathcal{D}$ and $\mathcal{D}_{aux}$ share no common entries, i.e. $\mathcal{D} \cap \mathcal{D}_{aux} = \emptyset$.
- Knowledge of the complete set of (l) potential outcomes for the sensitive attribute p.

Although the adversary only has black-box access to the actual model, they can utilize the received confidence score vectors to guide their pre-trained classifier, f_{adv}. This adversarial classifier learns from an auxiliary dataset, $\mathcal{D}_{aux}$, which is believed to share the same distribution as $\mathcal{D}$. The dataset $\mathcal{D}_{aux}$ consists of set of tuples $\{x_i, p_i, y_i\}_i$, representing public data, the sensitive attribute, and the prediction of the model.

The adversary's primary goal is to analyze responses from MLaaS to uncover users' concealed sensitive attributes. As part of this strategy, the adversary models f_{adv} using $\mathcal{D}_{aux}$, establishing the relationship $f_{adv} : (x, f(x)) \rightarrow p$.

A detailed visualization of these interactions is provided in Figure 1, illustrating the extent of the adversary's efforts to deduce sensitive information through systematic querying.

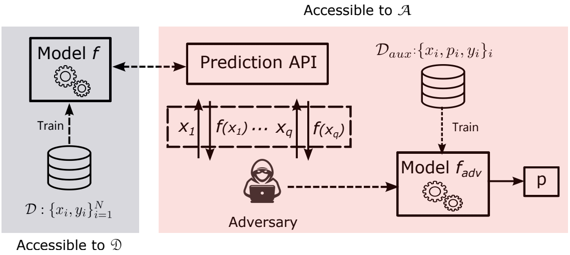
Accessible to $\mathcal{A}$
Model f
Prediction API
$\mathcal{D}_{aux}$: $\{x_i, p_i, y_i\}_i$
Train
Train
x_1 $f(x_1)$ ··· x_q $f(x_q)$
Model f_{adv}
p
Adversary
$\mathcal{D}: \{x_i, y_i\}_{i=1}^N$
Accessible to $\mathfrak{D}$

Figure 1: Attribute Inference Attack via MLaaS.

Our methodology centers on the assumption that an adversary can exploit biases and residual information memorized by the target classifier f_{target} to infer sensitive attributes. The adversary sets out to construct an inference classifier that uses as inputs the data point x and the prediction score vector provided by f_{target}, aiming to predict the value of the sensitive attribute. This strategy relies on utilizing a dataset $\mathcal{D}_{aux}$, carefully selected to mirror the distribution of the training set used to train f_{target}. By conducting supervised learning with $\mathcal{D}_{aux}$, where each record contains the corresponding value of the private attribute, the adversary aims to establish a correlation between the input data x, the prediction score vectors, and the values of the sensitive attribute. The effectiveness of this approach stems from the adversary's ability to identify and exploit specific prediction patterns and inherent biases in f_{target}, which are indicative of the sensitive attribute values.

3.3 Defense Mechanism

We introduce the third crucial entity of our model: the defender, denoted by $\mathfrak{D}$. Our proposal is based on the strategic perturbation of confidence score, deliberately introducing noise to confound and mislead adversarial classifiers. This requires meeting the following challenges:

1. Building upon our defensive strategy, the primary objective of our defense mechanism, denoted as $\mathcal{N}$, is to determine an optimal noise vector δ for addition to the prediction score vector υ, with the aim of mitigating the impact of sensitive attribute inference attacks. The mechanism $\mathcal{N}$ is designed to select a noise vector δ_i such that $\upsilon + \delta_i = \upsilon'$, when used by the adversary's classifier f_{adv}, results in a random prediction of attribute i. This is achieved by leveraging the probability μ_i to choose the noise δ_i, ensuring that, once added to the score vector to produce υ', f_{adv} is led to predict the attribute p_i randomly.
   This methodology represents a careful balancing act: it involves devising a mechanism that introduces enough randomness to confuse the adversary's classifier while maintaining the integrity and utility of the original score vector. The goal is for $\mathcal{N}$ to subtly alter the prediction output so that f_{adv} can no longer accurately infer the sensitive attribute, thus safeguarding user privacy. However, the challenge lies in accomplishing this without significantly compromising the quality or utility of υ, ensuring that the perturbed vector υ' continues to provide accurate and valuable predictions.
2. Within our framework, the defense confronts a significant hurdle: it does not have access to the adversary's classifier, f_{adv}. This limitation necessitates an inventive approach to counteract potential inference attacks effectively. As a solution, we propose the construction of our classifier, designated as f_{def}, to mirror the adversary's decision-making process. This model, developed through supervised machine learning techniques, aims to approximate the behavior and decision boundaries of f_{adv}, thereby serving as a proxy to anticipate and neutralize adversarial strategies.
   This strategy lies on the phenomenon of transferability, a well-documented characteristic in adversarial machine learning (Papernot et al., 2016). Transferability suggests that if a noise vector induces a misprediction in f_{def}, there is a high likelihood that the same noise vector will cause similar mispredictions in f_{adv}, assuming both classifiers have been trained on similar data distributions or share comparable decision boundaries. This premise underpins our defense tactic; by iterating over various noise vectors and evaluating their impact on f_{def}, we can identify those most likely to disrupt f_{adv} without necessitating direct access to the adversary's model.

4 A TWO-STAGE DEFENSE MODEL AGAINST AIA

The primary challenge in identifying noise vectors to perturb score vectors in machine learning models lies in the combinatorial explosion of parameters. Finding a noise vector δ to add to another vector υ to meet specific requirements dramatically increases the number of possibilities exponentially, making the implementation of a real-time optimization solution exceedingly difficult. Consequently, we considered categorizing noise vectors. Suppose there are k possible values for the sensitive attribute; we then have 10 categories of noise vectors, each designed to mislead the adversary's classifier into predicting a different sensitive attribute value.

To add another layer of complexity for the attacker, we select one of these vectors randomly according to a mechanism $\mathcal{N}$ based on certain probabilities, under the constraint that the utility of the perturbed vector υ' always predicts the same class, arg maxυ = arg maxυ'. The mechanism will have k different probabilities to pick a noise vector in each category. Our mechanism ensures that our defense strategy not only confuses the adversary but also maintains the predictive accuracy of the perturbed score vector.

4.1 Step 1: δ_i Determination

In this work, we begin with our confidence score vector. The objective is to identify elements to perturb such that the noise vector δ is determined. Our goal is to introduce noise by making minimal manipulations to this confidence score vector. This formulation closely mirrors an adversarial machine learning (ML) problem, where the primary aim is to subtly alter the input data to cause misclassification.

The constraints we consider ensure that the perturbed score vector remains a probability distribution, maintaining the essential characteristic that the sum of its elements equals one, and each element's value is between 0 and 1. Formally, the optimization problem is defined as:

$$\begin{aligned} & \delta^i = \arg\min_{\delta} \|\delta\|_2 \\ \text{subject to}\quad & f_{\text{adv}}(x, \upsilon + \delta^i) = i, \\ & \arg\max_j \upsilon = \arg\max_j \upsilon + \delta^i, \\ & \sum_{j=1}^{m} (\upsilon_j + \delta^i_j) = 1, \\ & 0 \leq (\upsilon_j + \delta^i_j) \leq 1, \quad \forall j = 1, \ldots, m. \end{aligned} \tag{2}$$

This optimization framework aims to pinpoint the optimal noise vector δ that satisfies these conditions, striking a balance between effectiveness in deceiving the classifier and adherence to the constraints that maintain the integrity of the probability distribution.

Traditional adversarial machine learning (ML) algorithms often fall short when faced with optimization problems that include constraints as detailed previously. The complexity introduced by these constraints, specifically the requirement for the perturbed score vector to remain a valid probability distribution, renders standard approaches less effective. This limitation highlights the need for innovative solutions tailored to handle such intricacies.

In our pursuit of a viable solution, we turn to a method proposed by Papernot *et al.*, known as the Jacobian-based Saliency Map Attack (JSMA). JSMA is an adversarial attack algorithm that identifies and perturbs a small subset of input features to significantly impact the output classification. The original JSMA algorithm focuses on the manipulation of input features based on their saliency, calculated by assessing the impact of each feature on the target classification. However, JSMA in its standard form does not inherently adhere to our constraint of maintaining the score vector as a probability distribution.

To bridge this gap, we propose an inspired variant of JSMA, tailored to our constrained optimization problem. Our approach modifies the JSMA methodology to incorporate the probability distribution constraint, ensuring that the perturbed score vector, when passed through a normalization function, remains a valid probability distribution. This adaptation allows us to explore the solution space more effectively while adhering to the essential constraints of our problem.

Therefore, we introduce a novel algorithm, NOISY (Noise Optimizer Inference Sensitive Yielder), designed to strategically navigate our optimization challenge. NOISY's core mission is to meticulously discover and apply an optimal perturbation vector δ^i to the existing confidence score vector υ. This vector is engineered not only to induce a specific prediction error within the adversary's classifier, aiming for the i^{th} value of the private attribute, but also to ensure that the resulting vector, when adjusted, adheres to the constraints of being a valid probability distribution. The algorithm operates through a series of iterative adjustments, carefully balancing the goal of causing the desired misclassification with the imperative to maintain the integrity of the original prediction. This is achieved by adhering to a rigorous constraint: the perturbed score vector must continue to predict the same class as it did prior to the perturbation. Through this approach, NOISY aims to achieve a delicate manipulation of the confidence scores, ensuring that while the adversary's classifier's accuracy is deliberately compromised, the utility and validity of the score vector remain uncompromised.

1. Calculating the gradient of the loss function with respect to the input vector to identify directions in which a perturbation would most likely lead to a misclassification.
2. Applying a constrained optimization to find δ^i that minimizes the perturbation under the set constraints.
3. Iteratively adjusting δ and verifying if the perturbed score vector $\upsilon + \delta$, after the normalization,

predicts the target attribute i.

Algorithm 1 shows our algorithm to find δ^i. NOISY's iterative nature allows for a refined search for the optimal perturbation, balancing the need to induce misclassification with the constraints of maintaining a probability distribution.

Algorithm 1: NOISY: Noise Optimizer Inference Sensitive Yielder.

```
Require: Confidence score vector $\upsilon$, classifier $f_{\text{def}}$, target attribute value $i$, step size $\alpha$, maximum iterations maxiter
Ensure: Perturbation vector $\delta^i$
Initialize iteration count $t = 0$
Initialize $\delta^i = \mathbf{0}$, $\upsilon' = \upsilon$
while $f_{\text{def}}(\text{softmax}(\upsilon' + \delta^i)) \neq i$ and $t <$ maxiter do
    // Identify the entries to modify based on saliency and constraints
    $e_{\text{inc}} = \text{argmax}_j \left\{ \frac{\partial f_{\text{def}}}{\partial x_j}(\upsilon') \middle| \delta^i_j = 0 \right\}$
    $e_{\text{dec}} = \text{argmax}_j \left\{ -\frac{\partial f_{\text{def}}}{\partial x_j}(\upsilon') \middle| \delta^i_j > 0 \right\}$
    // Modify the entries based on constraints
    $\delta_{e_{\text{inc}}} = \text{clip}(\delta^i_{e_{\text{inc}}} + \alpha, 0, 1)$
    $\delta_{e_{\text{dec}}} = \text{clip}(\delta^i_{e_{\text{dec}}} - \alpha, 0, 1)$
    // Adjust $\delta^i$ to maintain the original predicted class
    if $\arg\max_j(\upsilon_j) \neq \arg\max_j(\upsilon_j + \delta^i_j)$ then
        Reduce the magnitude of $\delta^i_{e_{\text{inc}}}$ and $\delta^i_{e_{\text{dec}}}$ to satisfy the constraint
    end if
    // Adjust $\delta^i$ to maintain $\upsilon' + \delta^i$ as a valid probability distribution
    total $= \sum_{j=1}^{m} (\upsilon'_j + \delta^i_j)$
    $\delta^i = \delta^i / \text{total}$
    Update $\upsilon' = (\upsilon + \delta^i)/\text{total}$
    $t = t + 1$
    Update $\upsilon' = \upsilon + \delta^i$
end while
return $\delta^i$
```

Our algorithm initializes with a zero perturbation vector δ^i and utilizes a saliency map to identify which elements to modify within the [0, 1] range, using a step size α. It carefully adjusts δ^i to ensure that the adjusted confidence score vector, $\upsilon' + \delta^i$, remains a valid probability distribution. The process iterates, focusing on modifying the confidence score vector to mislead the adversary's classifier into predicting a specific target class, while simultaneously ensuring that the original prediction class of the vector is preserved.

4.2 Step 2: $\mathcal{N}^*$ Determination

Upon concluding Step 1, we are equipped with k distinct categories of perturbation vectors, denoted as $\delta^1, \cdots, \delta^k$. In the Step 2, our goal is to construct a probability distribution that is the outcome of this mechanism. This distribution aims to be uniform (or 'flat') across the different perturbation vectors to introduce uncertainty into the adversary's choice. This uncertainty is crucial for ensuring that any selected noise vector maintains the classifier's prediction to the same class.

Our optimization problem can thus be formulated as the search for a probability distribution over the perturbation vectors that maximizes entropy, ensuring flatness, subject to the constraint of consistent class prediction. In mathematical terms, this problem is framed as:

$$\begin{aligned} \underset{\mu}{\text{maximize}} \quad & -\sum_{i=1}^{k} \mu_i \log \mu_i \\ \text{subject to} \quad & \underset{j}{\text{argmax}}\ \upsilon' = \underset{j}{\text{argmax}}\ \upsilon, \\ & \sum_{i=1}^{k} \mu_i = 1, \\ & \mu_i \geq 0,\ \forall i \in \{1, \ldots, k\}. \end{aligned} \tag{3}$$

To approach solving this optimization problem, we apply the Karush-Kuhn-Tucker (KKT) conditions, a fundamental method for solving constrained optimization problems. Initially, primal feasibility ensures that our solution adheres to all established constraints, maintaining the integrity of our problem's formulation. Then, stationarity is achieved when we identify appropriate Lagrange multipliers—λ for equality constraints and ν_i for inequality constraints—such that the gradient of the Lagrangian with respect to μ vanishes at the optimum point. This guarantees that our solution is not only feasible but also optimally aligned with our objective function and constraints. Dual feasibility requires that the Lagrange multipliers associated with our inequality constraints are non-negative, a condition ensuring that our solution resides within the permissible solution space. Lastly, complementary slackness insists that for each inequality constraint, the product of its Lagrange multiplier and the constraint itself equals zero at the optimum, blending the boundary between feasibility and optimality. Together, these conditions meticulously guide us to a solution that is not only within bounds but also optimal, ensuring a rigorous adherence to both our problem's structure and its inherent constraints.

$$L(\mu,\lambda,\nu) = -\sum_{i=1}^{k}\mu_i \log\mu_i + \lambda\left(\sum_{i=1}^{k}\mu_i - 1\right) + \sum_{i=1}^{k}\nu_i(-\mu_i) \tag{4}$$

Our optimization strategy employs the Lagrangian $L(\mu,\lambda,\nu)$ to maximize entropy in the perturbation vector distribution μ, under specific constraints. The objective, $-\sum_{i=1}^{k}\mu_i\log\mu_i$, seeks a uniform distribution across k vectors, enhancing unpredictability. The term $\lambda(\sum_{i=1}^{k}\mu_i - 1)$ ensures the distribution's normalization, while $\sum_{i=1}^{k}\nu_i(-\mu_i)$ imposes non-negativity on each μ_i, with λ and ν_i as Lagrange multipliers for equality and inequality constraints, respectively.

4.2.1 Practical Interpretation of $\mathcal{N}^*$

The mechanism $\mathcal{N}$ plays a pivotal role in our optimization framework, serving as the strategic core for selecting the optimal noise vector under tightly defined constraints. This mechanism is designed to navigate through the complex landscape of adversarial perturbations, aiming to identify a perturbation strategy that not only adheres to operational constraints—such as maintaining the classifier's prediction—but also introduces a level of indeterminacy and diversity in the adversarial examples generated.

In practical terms, $\mathcal{N}$ determines the distribution of probability across various noise vectors $(\delta^1,\ldots,\delta^k)$ in a manner that keeps the classifier's output consistent, yet makes the adversary's actions less discernible. By doing so, $\mathcal{N}$ effectively increases the difficulty for defensive mechanisms to predict and mitigate these adversarial perturbations, securing a strategic advantage.

5 EXPERIMENTATION

In this section, we discuss our experimental framework employed to validate the effectiveness of our proposed security mechanism against attribute inference attacks. The core objective of our investigation is to substantiate that our model can significantly deceive the attacker's classifier, thereby safeguarding sensitive attributes deductible from the score vector. Simultaneously, we aim to preserve the inherent utility of the score vector for legitimate purposes. This dual achievement is facilitated through strategic perturbations introduced to the confidence scores.

5.1 Dataset and Setup

Texas-100X (Jayaraman, 2022)**:** the data set we employ, termed Texas-100X, serves as an expanded version of the Texas-100 hospital dataset, previously introduced by Shokri *et al.* (Shokri et al., 2017). Each entry in this dataset provides comprehensive demographic details of patients—spanning from age, gender, and ethnicity—to nuanced medical data like the length of hospital stays, mode of admission, diagnostic reasons, patient outcomes, incurred charges, and primary surgical interventions. The objective set for this dataset is to anticipate one out of 100 possible surgical interventions, grounded on individual health records.

While the predecessor, Texas-100, comprised 60,000 entries with 6,000 obscured binary attributes, the Texas-100X dataset contains an impressive 925,128 patient records gathered from 441 distinct hospitals. Specially, this dataset retains the primary 10 demographic and medical traits in their original, decipherable state.

Census19: the Census19 dataset (cen, 2019) is a modern extension of the well-known Adult dataset (Asuncion and Newman, 2007), derived from the 1994 Census data. While the original housed around 48,000 records with 14 attributes, Census19 version pulls from the U.S. Census Bureau's 2019 database, offering 1,676,013 entries with 12 pivotal attributes.

These records, organized based on Public Use Microdata Areas (PUMAs), capture key demographic aspects of U.S. residents: age, gender, race, marital status, education, occupation, work hours, country of origin, and certain disability indicators. The primary classification challenge with Census19 is to determine whether an individual's annual income surpasses $90,000 an inflation-adjusted figure from the Adult dataset's $50,000 threshold.

In our evaluations involving both Texas-100X and Census19, we randomly pick 50,000 entries to establish the training dataset and employ it to train a two-layer neural network. Additionally, we isolate another 25,000 distinct records from the leftover data to constitute the testing dataset, ensuring no overlap between the training and test datasets.

The neural network employed for our defense model is structured as follows:

- **Input Layer:** configured to match the dimensionality of the feature space in the Texas-100X and Census19 datasets. This ensures that the network can process each input attribute without loss of information.
- **Hidden Layers:** comprises multiple layers to enhance the model's ability to capture nonlinear relationships within the data. Each layer is equipped with a ReLU activation function to introduce nonlinearity, facilitating complex decision boundary formations essential for effective defense.
- **Output Layer:** the final layer is designed to output the perturbed score vector. The dimension-

ality of this layer corresponds to the number of classes in the dataset, with a softmax activation function applied to convert the network's output into a probability distribution over potential classes.

5.1.1 Defense Mechanism Integration

The core of our defense strategy involves the $\mathcal{N}^*$ determination mechanism, which dynamically introduces perturbations into the confidence score vector. This neural network architecture is pivotal in evaluating the impact of such perturbations, allowing for real-time adjustments to ensure that the perturbed vector deceives the attacker's classifier while preserving the integrity and utility of the original score vector.

To implement the $\mathcal{N}^*$ mechanism, the network is trained on adversarially perturbed data alongside clean data, optimizing for two primary objectives: minimizing the success rate of attribute inference attacks and maintaining high accuracy on legitimate classification tasks. This dual-objective training regimen is instrumental in hardening the defense model against sophisticated adversarial strategies.

5.1.2 Training and Evaluation

The model undergoes rigorous training using a curated dataset that amalgamates samples from both Texas-100X and Census19, ensuring comprehensive exposure to diverse data representations. The training process leverages a cross-entropy loss function, which is effective for classification tasks and facilitates the optimization of the network's weights in the context of our defense objectives.

5.1.3 Evaluation Metrics

In our evaluation, we aim to conduct a comprehensive comparison of our two-step model against two established privacy-preserving techniques: Local Differential Privacy (LDP) (Avent et al., 2017) and k-Anonymity (Zhao et al., 2018). These methods are well-regarded for their ability to mitigate the risks associated with sensitive attribute inference attacks within Machine Learning as a Service (MLaaS) environments, each utilizing distinct mechanisms that impact data utility and processing efficiency in unique ways.

Our two-step model leverages sophisticated adversarial examples and a strategic selection mechanism to preserve privacy against Attribute Inference Attacks (AIA). It will be assessed alongside LDP rather than standard Differential Privacy due to LDP's suitability for environments where the central aggregation of data is not feasible or where the trust in a central curator is limited. LDP applies controlled statistical noise directly at the data source, masking individual contributions before the data aggregation occurs. This method enables greater privacy assurance directly on the user's device without requiring trust in the central server's handling of their data. An ε parameter, randomly selected from the interval $[0,10]$ (Avent et al., 2017), will be tuned to balance privacy protection and the utility of predictions.

Conversely, k-Anonymity protects privacy by ensuring each record in a dataset is indistinguishable from at least $k-1$ other records with similar attributes. We will choose a k value that maximizes the difficulty of associating data with specific individuals while maintaining sufficient data granularity for meaningful analysis.

The evaluation focuses on three critical metrics: the rate of successful inference attacks, the impact on confidence score utility, and the computational speed of each method. By measuring the inference rate, we aim to understand how well each method conceals sensitive attributes from potential attackers. The utility loss metric will help us gauge the extent to which the protection method affects the data's usefulness for legitimate analytical tasks. Finally, computational speed will be assessed to determine the efficiency and practicality of implementing each method in real-world scenarios. This comparative analysis will not only highlight the strengths and weaknesses of our model but also contribute valuable insights into the trade-offs involved in implementing privacy-preserving techniques in data-driven applications.

5.2 Results and Analysis

Figure 2 illustrates the comparative analysis of sensitive attribute inference rates across our developed model, the ”Two-Step Adversarial Defense”, Local Differential Privacy (LDP), and k-Anonymity methods. Our ”Two-Step Adversarial Defense” method consistently shows the lowest inference rates, indicating its effectiveness in minimizing the likelihood of sensitive attributes being accurately inferred by adversaries. This performance highlights the advantages of our approach in enhancing data privacy compared to both LDP and k-Anonymity.

In Figure 3, we examine the impact of each privacy-preserving method on classification error rates. Our ”Two-Step Adversarial Defense” method not only provides substantial protection against attribute inference attacks but does so with minimal impact on the classification accuracy.This result under-scores our method's capability to maintain data utility for legitimate analytical purposes while providing robust defense mechanisms. It effectively balances security with data utility, outperforming both Local Differential Privacy (LDP) and k-Anonymity in this regard.

The Figure 4 focuses on the efficiency of generating noise vectors, a critical aspect in the practical application of privacy-preserving techniques. Our ”Two-Step Adversarial Defense” method is demonstrated to generate noise vectors more swiftly than both LDP and k-Anonymity. This suggests that our method not only enhances privacy protection but also does so more efficiently. Such efficiency makes our proposed method particularly suitable for environments where rapid data processing is essential, thus offering significant advantages over the compared methodologies.

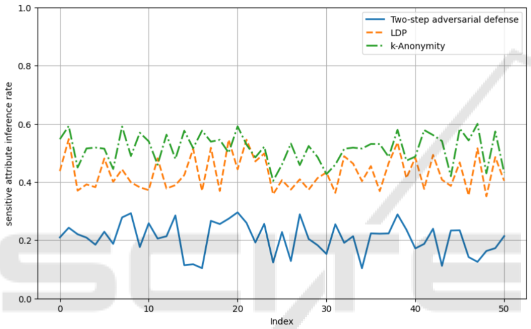

Figure 2: Comparison of Sensitive Attribute Inference Rate between Two-Step Adversarial Defense, LDP and k-Anonymity.

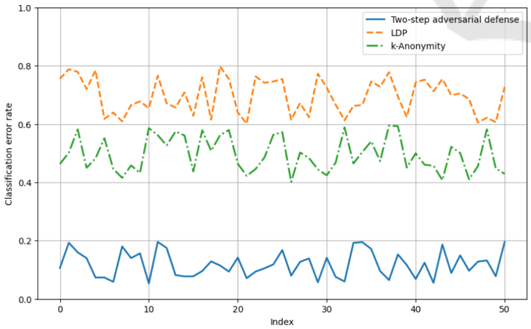

Figure 3: Evaluation of the Impact on Classification Error Rate: Two-Step Adversarial Defense vs. Privacy Methods.

In each figure, our ”Two-Step Adversarial Defense” method consistently surpasses its counterparts, demonstrating comprehensive advantages in protecting sensitive information, preserving data utility, and ensuring operational efficiency. These results affirm the effectiveness of our approach in balancing robust privacy protection with the practical demands of real-world applications.

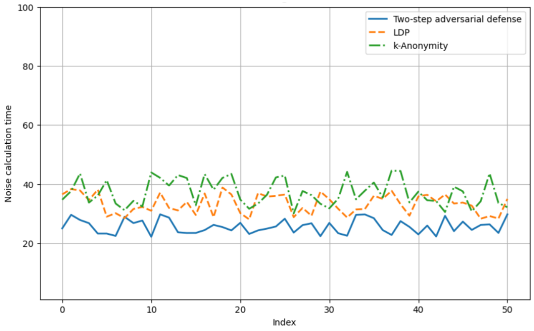

Figure 4: Noise Generation Speed Performance: Comparative Analysis between Two-Step Adversarial Defense, LDP and k-Anonymity.

6 CONCLUSION

In this study, we developed and evaluated the ”Two-Step Adversarial Defense” method to enhance privacy in MLaaS environments susceptible to attribute inference attacks. Our approach, which introduces sophisticated adversarial examples followed by strategic noise vector selection, has proven effective in reducing the likelihood of sensitive attribute exposure while maintaining the utility of the data for legitimate analytical purposes.

Moving forward, we plan to further enhance our model by incorporating strategic concepts from game theory into the noise vector selection process. This adjustment will allow for a more calculated and context-aware application of noise, potentially increasing the robustness of our privacy protections. Additionally, we will expand our comparative analysis with existing privacy-preserving methods. This expanded comparison will provide a clearer understanding of the impact our proposed approach has in various operational contexts, helping to refine our strategies and solidify our defenses against evolving threats to sensitive data.

REFERENCES

(2019). Census19 data set. https://www.census.gov/programs-surveys/acs/ Accessed: 2023-08-24.

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

Asuncion, A. and Newman, D. (2007). Uci machine learning repository.

Ateniese, G., Mancini, L. V., Spognardi, A., Villani, A., Vitali, D., and Felici, G. (2015). Hacking smart machines with smarter ones: How to extract meaningful data from machine learning classifiers. *International Journal of Security and Networks*, 10(3):137–150.

Avent, B., Korolova, A., Zeber, D., Hovden, T., and Livshits, B. (2017). {BLENDER}: Enabling local search with a hybrid differential privacy model. In *26th USENIX Security Symposium (USENIX Security 17)*, pages 747–764.

Bouhaddi, M. and Adi, K. (2023). Mitigating membership inference attacks in machine learning as a service. In *2023 IEEE International Conference on Cyber Security and Resilience (CSR)*, pages 262–268. IEEE.

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

Chen, J., Li, K., and Philip, S. Y. (2021). Privacy-preserving deep learning model for decentralized vanets using fully homomorphic encryption and blockchain. *IEEE Transactions on Intelligent Transportation Systems*, 23(8):11633–11642.

du Pin Calmon, F. and Fawaz, N. (2012). Privacy against statistical inference. In *2012 50th annual Allerton conference on communication, control, and computing (Allerton)*, pages 1401–1408. IEEE.

Dunis, C., Middleton, P. W., Karathanasopolous, A., and Theofilatos, K. (2016). *Artificial intelligence in financial markets*. Springer.

Dwork, C., Roth, A., et al. (2014). The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407.

Farnadi, G., Sitaraman, G., Sushmita, S., Celli, F., Kosinski, M., Stillwell, D., Davalos, S., Moens, M.-F., and De Cock, M. (2016). Computational personality recognition in social media. *User modeling and user-adapted interaction*, 26:109–142.

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

Gong, N. Z. and Liu, B. (2016). You are who you know and how you behave: Attribute inference attacks via users' social friends and behaviors. In *25th USENIX Security Symposium (USENIX Security 16)*, pages 979–995.

Hildebrandt, M. (2018). Law as computation in the era of artificial legal intelligence: Speaking law to the power of statistics. *University of Toronto Law Journal*, 68(supplement 1):12–35.

Hu, H., Salcic, Z., Sun, L., Dobbie, G., Yu, P. S., and Zhang, X. (2022). Membership inference attacks on machine learning: A survey. *ACM Computing Surveys (CSUR)*, 54(11s):1–37.

Jayaraman, B. (2022). Texas-100x data set. https://github.com/bargavj/texas100x. Accessed: 2023-08-24.

Jayaraman, B. and Evans, D. (2019). Evaluating differentially private machine learning in practice. In *28th USENIX Security Symposium (USENIX Security 19)*, pages 1895–1912.

Jayaraman, B. and Evans, D. (2022). Are attribute inference attacks just imputation? In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 1569–1582.

Jia, J. and Gong, N. Z. (2018). {AttriGuard}: A practical defense against attribute inference attacks via adversarial machine learning. In *27th USENIX Security Symposium (USENIX Security 18)*, pages 513–529.

Jia, J., Salem, A., Backes, M., Zhang, Y., and Gong, N. Z. (2019). Memguard: Defending against black-box membership inference attacks via adversarial examples. In *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security*, pages 259–274.

Kosinski, M., Stillwell, D., and Graepel, T. (2013). Private traits and attributes are predictable from digital records of human behavior. *Proceedings of the national academy of sciences*, 110(15):5802–5805.

Linden, G., Smith, B., and York, J. (2003). Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, 7(1):76–80.

Liu, Y., Chen, X., Liu, C., and Song, D. (2016). Delving into transferable adversarial examples and black-box attacks. *arXiv preprint arXiv:1611.02770*.

Mahajan, A. S. D., Tople, S., and Sharma, A. (2020). Does learning stable features provide privacy benefits for machine learning models. In *NeurIPS PPML Workshop*.

Malekzadeh, M., Borovykh, A., and Gündüz, D. (2021). Honest-but-curious nets: Sensitive attributes of private inputs can be secretly coded into the classifiers' outputs. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 825–844.

Mehnaz, S., Dibbo, S. V., De Viti, R., Kabir, E., Brandenburg, B. B., Mangard, S., Li, N., Bertino, E., Backes, M., De Cristofaro, E., et al. (2022). Are your sensitive attributes private? novel model inversion attribute inference attacks on classification models. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 4579–4596.

Papernot, N., McDaniel, P., and Goodfellow, I. (2016). Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *arXiv preprint arXiv:1605.07277*.

Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., and Swami, A. (2017). Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pages 506–519.

Rivest, R. L., Adleman, L., Dertouzos, M. L., et al. (1978). On data banks and privacy homomorphisms. *Foundations of secure computation*, 4(11):169–180.

Salamatian, S., Zhang, A., du Pin Calmon, F., Bhamidipati, S., Fawaz, N., Kveton, B., Oliveira, P., and Taft, N. (2015). Managing your private and public data: Bringing down inference attacks against your privacy. *IEEE Journal of Selected Topics in Signal Processing*, 9(7):1240–1255.

Shokri, R. (2014). Privacy games: Optimal user-centric data obfuscation. *arXiv preprint arXiv:1402.3426*.

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

Shokri, R., Theodorakopoulos, G., and Troncoso, C. (2016). Privacy games along location traces: A game-theoretic framework for optimizing location privacy. *ACM Transactions on Privacy and Security (TOPS)*, 19(4):1–31.

Song, C. and Shmatikov, V. (2019). Overlearning reveals sensitive attributes. *arXiv preprint arXiv:1905.11742*.

Weinsberg, U., Bhagat, S., Ioannidis, S., and Taft, N. (2012). Blurme: Inferring and obfuscating user gender based on ratings. In *Proceedings of the sixth ACM conference on Recommender systems*, pages 195–202.

Weiss, J. C., Natarajan, S., Peissig, P. L., McCarty, C. A., and Page, D. (2012). Machine learning for personalized medicine: predicting primary myocardial infarction from electronic health records. *Ai Magazine*, 33(4):33–33.

Yeom, S., Giacomelli, I., Fredrikson, M., and Jha, S. (2018). Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st computer security foundations symposium (CSF)*, pages 268–282. IEEE.

Zhao, P., Jiang, H., Wang, C., Huang, H., Liu, G., and Yang, Y. (2018). On the performance of *k*-anonymity against inference attacks with background information. *IEEE Internet of Things Journal*, 6(1):808–819.