femington

Int. J. Intell. Syst. Data Sci. (2026) 1(4):1–15
Article ID: IJISDS-2026-363

Training Dataset Exposure Through Adversarial Model Query Analysis: A Black-Box Privacy Breach Framework

Pramod Prakash
pramodprakash16@gmail.com
Independent Researcher

Abstract

Machine-learning-as-a-service (MLaaS) platforms let organizations deploy predictive models without revealing their internal architecture, but this opacity does not guarantee the privacy of the data used to train them. This paper presents a systematic framework for membership inference attacks, in which an adversary with only black-box query access to a deployed classifier determines whether a specific individual's record was part of its training set. We introduce a shadow-model methodology that trains auxiliary classifiers on synthetically generated data produced via model-based synthesis, known population statistics, or noisy real data to learn the behavioral differences a target model exhibits on seen versus unseen inputs. Evaluated against commercial platforms including Google Prediction API and Amazon ML across seven datasets spanning healthcare, retail, location, and image domains, our attacks achieve median precision of 0.657 and 0.678 on the two platforms, respectively, with healthcare records showing vulnerability of approximately 65.7% precision. We identify overfitting, output granularity, and class imbalance as primary drivers of leakage, and evaluate mitigations such as output truncation, temperature scaling, and regularization, finding that each offers only partial protection at the cost of utility. These findings expose significant gaps in current regulatory frameworks such as GDPR and HIPAA regarding inference-based privacy violations.

Keywords

• Privacy • Machine Learning • Membership Inference • Black-Box Attack • Shadow Models • Data Leakage

1. Introduction

The rise of platforms offering machine learning as a service (MLaaS) has made sophisticated AI capabilities available to many organizations, allowing them to deploy complex models without in-depth algorithms expertise. Offered by some of the biggest technology companies like Google, Amazon, and Microsoft, these platforms work as a black box in which customers upload datasets and receive API endpoints to query the model. The internal architecture, training procedures, and hyperparameters, in this case, remain a mystery. Although this abstraction makes it easier for people to adopt machine learning and eases technical entry barriers, it creates major security blind spots regarding the leakage of information from trained models [1, 2]. This study is concerned with a major threat in this network: an external adversary observing the outputs of a model and determining whether a certain person's records were in the training dataset.

Membership inference attacks pose a serious privacy risk that cannot be offset with standard data protection measures. Membership inference seeks to determine if a known record was part of the model's

secret training data, in contrast to model inversion techniques which construct approximate training data. The ability to know the participation of a dataset in a model can leak sensitive knowledge. Such ability is concerning in case of healthcare, finance, and government services, as even the knowledge of participation can reveal sensitive input attributes. As an example, identifying that a certain patient's medical record was used to train a disease prediction model reveals the patient's health status, which may violate the HIPAA and GDPR [3]. The issue is problematic in ML systems that are based on the cloud, which expose the model through a public API. As a result, there is a possible attack surface for any entity connected to the net [4].

Our study provides a systematic framework for measuring and exploiting this attack in various practical situations. We show membership inference is not only a theoretical problem, but also a practical attack with high success rates against production systems. The central insight underlying our approach is that machine learning models, especially overfitting ones, behave in a measurable manner differently when processed with inputs from their training distribution than new inputs from the same population. The differences in behavior generate statistical signatures that can be learned and leveraged by secondary attack models, despite having no access to the internal workings of the target model [5, 6].

The importance of our contribution lies in the specific framework that integrates adversarial knowledge settings, cross-dataset evaluation across seven diverse datasets, query-efficiency analysis, and a systematic mitigation analysis. While shadow-model techniques have been established in prior membership inference literature, our contribution is the unified evaluation methodology that enables direct comparison across datasets, the analysis of query budgets in realistic adversarial scenarios, and the characterization of defense effectiveness across different threat models. This framework provides actionable insights for deploying privacy-preserving machine learning in practice. Our baseline vulnerability evaluation of today's MLaaS systems is conducted over seven heterogeneous datasets, namely, hospital discharge records, retail transactions, location traces, and image data. The seven datasets are Purchase-100, Hospital-100, Location-30, CIFAR-10, CIFAR-100, Adult, and MNIST, and parameter-sensitivity subsets are not counted as separate datasets. Our results show very high success rate, with median accuracy of 0.657 against Google's Prediction API and 0.678 against Amazon Machine Learning for purchase classification tasks. As it stands, organizations' practices in deploying ML models do not afford adequate privacy guarantees, creating gaps in regulatory compliance and exposure to liability [7, 8].

In addition to the attack methodology, we investigate various factors that could cause a vulnerability. Such factors include model overfitting, class imbalance, and granularity of the output. In this work, we investigate two mitigations as output truncation and differential privacy and quantify their efficacy when it comes to leakage. The paper recommends the platform providers and model user, through which they can achieve model utility along with a privacy protection. We aim to enhance machine learning practices in privacy engineering through reproducibility structures of attack and measurable security metrics.

2. Related Work

The growing body of literature on AI security and privacy has established that modern machine learning systems are vulnerable to a wide range of inference attacks, even when they are deployed as opaque black boxes. Early surveys on data leakage have cataloged both direct and indirect channels through which sensitive information can be exfiltrated [7, 8]. More recent work has systematically analyzed the privacy risks introduced by the widespread adoption of machine learning, highlighting that model outputs themselves can inadvertently reveal training data properties [1, 2]. In parallel, researchers have explored the role of deep learning in ensuring privacy integrity and security, proposing architectures that aim

to mitigate such risks [3]. A position paper by Thuraisingham [4] further argues that while AI can be harnessed for good, it also introduces novel attack vectors that must be addressed proactively.

A particularly relevant line of research focuses on black-box attacks against machine learning models, where the adversary has only query access to the target system. Papernot et al. [5] demonstrated that practical black-box attacks can be mounted using transferability and synthetic data generation, a concept that underpins the shadow model methodology. Similarly, Alzantot et al. [6] proposed gradient-free optimization techniques for crafting adversarial examples and performing membership inference, showing that effective attacks are feasible even without gradient information. These foundational works have paved the way for more systematic evaluations of privacy leakage in deployed ML systems. Complementary to these, Guo et al. [9] explored shadow-based techniques in a different context image shadow removal illustrating that the "shadow" concept appears across various subfields of AI and can be repurposed for both benign and adversarial goals.

Beyond general attacks, several studies have investigated privacy and security in specific application domains. For instance, Penumarthi [10] examined model-driven engineering for health data management, underscoring the sensitivity of medical records and the need for robust privacy controls. In the realm of instant messaging, Shaik [11] evaluated cryptographic frameworks, while Kuruppathukattil [12] addressed secure data transmission in cloud environments, both highlighting the importance of protecting data in transit and at rest. On the regulatory side, Chevva [13] discussed the challenges of balancing accuracy and efficiency in distributed ML systems, along with the regulatory implications of privacy breaches, a theme that resonates with our findings on GDPR and HIPAA gaps.

Privacy-preserving techniques have also been actively explored. The work on federated averaging for IoT sensor networks [14] and decentralized intelligence for smart cities [15] demonstrate efforts to limit data centralization and reduce exposure. However, these approaches introduce new challenges in communication efficiency and leakage control, as also noted in broader federated learning research. Device fingerprinting for authentication [16] and cloud system observability [17] further illustrate the breadth of security considerations in modern distributed infrastructures.

The diversity of machine learning applications introduces varied privacy risks. For example, bibliometric clustering for citation networks [18], event-trace cohesion in collaborative editing [19], and generative modeling for symbolic audio [20] each present unique data handling challenges. Similarly, performance optimization in big data streaming [21], high-performance data stores [22], and LSM tree operations [23] highlight that infrastructure decisions can influence data exposure. Even seemingly unrelated domains such as agricultural crop recommendation [24] and social media analytics [25] must consider privacy implications when training models on user data. Moreover, data tier architectures for enterprise SaaS [26] and tuple-indexed labeling for graph metadata [27] illustrate that data organization choices can affect both performance and the potential for inference leakage. Finally, code anomaly detection [28] and semantic enrichment of MathML [29] reinforce the notion that every ML deployment must be evaluated for potential inference attacks.

Overall, the related work underscores that membership inference is not an isolated phenomenon but a systemic risk that spans algorithms, platforms, and application domains. Our work builds on these foundations by providing a unified framework for measuring and mitigating this risk in production MLaaS environments.

3. Threat Model and Problem Formulation

We imagine a believable threat where an adversary wishes to determine whether certain individual records were part of the training data of some deployed machine learning prediction service offered by a cloud provider. The limitations on capabilities of adversary are designed to reflect realistic attack conditions on production ML systems. The target model is a classification system based on a public API that accepts input records and returns a prediction vector, indicating the probabilities of classes. The adversary does not know the model's internal structure, its training algorithm, its hyperparameters, or the training dataset's composition, other than that of the domain [3].

The adversary has black-box query access to the target model, which means he can submit arbitrary input records according to the feature format expected from the model and receive the corresponding prediction outputs. This access model corresponds directly to common usage patterns of MLaaS models as they are deployed as web services with consistent APIs [5]. Depending on the projected attack scenario, the adversary may also have auxiliary information about the data domain, namely distributions, ranges and semantics. However, the models do not have direct access to the private training data itself or to any explicit statistical summaries of it. The adversary wishes to perform a binary classification for a record of interest, check whether the record was part of the model's training dataset (member) or not (non-member).

Formally, let $f_{\text{target}} : \mathcal{X} \to \mathcal{Y}$ represent the target classification model, where $\mathcal{X}$ denotes the input space and $\mathcal{Y} = \{\mathbf{y} \in [0,1]^c : \sum_{i=1}^{c} y_i = 1\}$ represents probability vectors over c classes. The training dataset $D_{\text{train}} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n} \subset \mathcal{X} \times \{1, \ldots, c\}$ consists of n labeled records. For a query record $\mathbf{x}^*$ with known true label y^*, the adversary observes the prediction vector $\mathbf{y}^* = f_{\text{target}}(\mathbf{x}^*)$. The attack succeeds if the adversary can correctly infer the membership status $m^* \in \{\text{IN}, \text{OUT}\}$ where $m^* = \text{IN}$ if $(\mathbf{x}^*, y^*) \in D_{\text{train}}$ and $m^* = \text{OUT}$ otherwise.

There are a number of realistic use cases in the attack scenario. Let us take a healthcare application first where hospitals jointly train a model on patient records to predict disease outcome or treatment efficacy. Should these models be made available to outside researchers or used by outside applications, adversaries could test them to see if their patients' records were part of training data. This is particularly alarming, given that adversaries are interested in very particular data for examples, students diagnosed with very precisely defined health conditions [10]. Another growing use of ML is in financial institutions for fraud detection and credit scoring. Membership inference might reveal whether individual transaction histories were used to build the models, thus revealing their financial status and/or prior fraudulent activity. Moreover, government entities are deploying ML systems to automate member services, which can infringe on citizen privacy through membership inference in sensitive datasets [4].

The adversary's background knowledge falls into three categories according to accessibility: (1) model-based synthesis, only accessible through API, (2) statistics-based synthesis, wherein the population marginals are known, and (3) noisy real data, where there exist similar but dissimilar data-sets. We assess all three cases to ensure a comprehensive assessment. The effectiveness of an attack is evaluated by assessing whether its accuracy is significantly higher than random guessing (50%), using standard classification measures like precision, recall, and F1-score.

To keep things relevant, we avoid stronger attacks that are not in our threat model. The model is not assumed to be over queried by the adversary as production systems usually have limitations and monitoring in place. We also dismiss attacks that take advantage of the access to model parameters and/or inner workings of the training algorithm. We also assume the adversary is not able to interfere with the training phase, through data poisoning or any other means of active attacks [1]. This conservative approach makes sure that our findings reflect theses in current M.L. deployment.

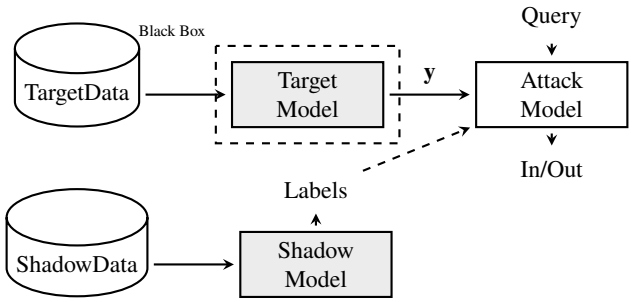

Figure 1: Black-box membership inference attack architecture using shadow models.

4. Shadow Model Attack Methodology

The core innovation of our approach is the shadow model technique, which enables effective attack model training without access to the target model's internal details or training data. This methodology transforms the membership inference problem into a standard binary classification task by creating proxy training data that mimics the target model's behavior [2]. The fundamental insight is that multiple models trained on similar data distributions using the same learning algorithm exhibit comparable behavioral patterns when distinguishing between training and non-training inputs. By constructing an ensemble of such shadow models and recording their prediction behaviors on inputs with known membership status, we create a labeled dataset for training the attack classifier.

The specific contribution of this work lies in the comprehensive framework that integrates adversarial knowledge settings, cross-dataset evaluation across seven diverse datasets, query-efficiency analysis, and a systematic mitigation analysis. While shadow-model techniques have been established in prior membership inference literature, our contribution is the unified evaluation methodology that enables direct comparison across datasets, the analysis of query budgets in realistic adversarial scenarios, and the characterization of defense effectiveness across different threat models. This framework provides actionable insights for deploying privacy-preserving machine learning in practice.

The attack methodology comprises four systematic phases: shadow data generation, shadow model training, attack data collection, and attack model construction. In the shadow data generation phase, the adversary creates synthetic datasets that approximate the distribution of the target model's training data. We develop and evaluate three distinct generation strategies with varying assumptions about adversary knowledge. The model-based approach requires only black-box query access to the target model, using hill-climbing optimization to discover input records that elicit high-confidence predictions, which are statistically likely to resemble training distribution samples. The statistics-based approach utilizes known marginal feature distributions from the population, independently sampling each feature to construct synthetic records. The noisy real data approach assumes access to datasets from similar domains or contexts, applying controlled randomization to simulate distributional differences [6].

Once shadow datasets are generated, the adversary trains multiple shadow models using the same MLaaS platform or algorithm as the target model. This ensures behavioral similarity despite different training data compositions. Each shadow model is trained on a distinct dataset with known membership labels for all candidate records. The adversary then queries each shadow model with two sets of inputs: its own training data (labeled IN) and a disjoint test set of equal size (labeled OUT). The prediction vectors from these queries, combined with the true class labels, form the feature representation for attack model training. This process yields a comprehensive dataset capturing how similar models behave when processing memorized versus novel inputs [5].

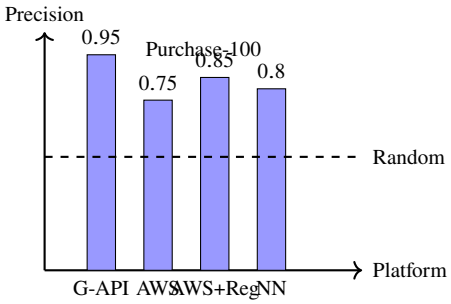

Figure 2: Membership inference attack precision across ML platforms on the Purchase-100 dataset.

The architecture for the attack model captures subtle differences in behavior present in predictions. We learn a set of attack models, one for each output class of the target model rather than a single classifier. This specialization enhances correctness since the distributions of prediction vectors differ markedly across classes. The input to each attack model consists of a concatenated vector, which includes the true class label and the complete prediction vector of the target model. Output is one of two binary values (IN/OUT) demonstrating simplest membership probabilities. Through practical observation, we established that these neural networks, which consists of one hidden layer offers superb performance for this task while also remaining simple and interpretable [4].

The success of the shadow training method depends on the transferability of behaviors in models trained under the same conditions. According to our experiments, this assumption is true across datasets and models. Interestingly, attack models designed to mimic the behavior of shadow models work well with the target model, even though the target and shadow model do not share any training samples. The transferability of this attack reveals that the vulnerability of membership inference exists due to properties of the algorithm rather than the dataset. Even with the addition of a small number of shadow models (10–20), the approach remains effective. This limits the computational cost against production systems.

Special attention must go to algorithmic details of model-based synthesis as they facilitate minimal-knowledge attacks. The hill-climbing technique makes iterative changes to candidate records to maximize the confidence of a prediction for the target class. It uses adaptive step lengths with an initial step size of 0.1 times the feature range, adjusted by a factor of 0.95 after each unsuccessful iteration. Convergence is declared when the confidence improvement over the last 50 iterations is less than 10^{-4}, or after a maximum of 2000 iterations per candidate record. When the confidence exceeds a minimum threshold of 0.8 and the margin above the next highest class probability exceeds 0.1, records are probabilistically accepted into the synthetic dataset with probability equal to the confidence score. In practice, this approach can efficiently search high-dimensional input spaces, although it struggles with complicated domains (e.g., high-resolution images). When faced with these instances, choices include a statistics-based approach or a noisy data-based approach.

The noisy real data approach assumes access to datasets from similar domains or contexts, applying controlled randomization to simulate distributional differences. For continuous features, we add independent Gaussian noise $\mathcal{N}(0, \sigma^2)$ where $\sigma = 0.1 \cdot \text{range}(feature)$. For binary features, we flip each bit independently with probability $p_{\text{flip}} = 0.1$. For categorical features with k categories, we replace the true category with a uniformly random alternative category with probability $p_{\text{corrupt}} = 0.15$. These noise levels were chosen to preserve gross distributional properties while obscuring exact record identities.

The strength of the attack method lies in its flexibility against different knowledge scenarios. Using only the given access via API, model-based synthesis can produce enough training data to launch

successful attacks. The quality of synthetic data will improve when the population statistics are known. Noisy data approaches achieve near optimal performance when using similar datasets. This flexibility means that the threat model is widely applicable in the real world, ranging from unknown targets to known targets. The modular design of these methodologies also allows for integration of alternative data generation or model training approaches as developed.

5. Experimental Evaluation

We perform extensive experiments on seven diverse datasets to test the membership inference vulnerability of models in practice. The seven datasets are Purchase-100, Hospital-100, Location-30, CIFAR-10, CIFAR-100, Adult, and MNIST, with parameter-sensitivity subsets not counted as separate datasets. We analyse both a locally-trained neural network and a neural network hosted on a commercial MLaaS platform for vulnerability factors. The methodology applied in all experiments is consistent. Target models are trained on a randomly selected subset of instances; shadow models are constructed using one of three ways to generate data; attack models are trained on shadow model behaviour; and evaluation metrics are computed on queries to a held-out target model. Having systematic methodologies allows reasonable comparisons across configurations and identification of relevant vulnerability factors.

The selected datasets are from different domains and varied characteristics. CIFAR-10 and CIFAR-100 are image classification benchmarks with 10 and 100 classes respectively. Kaggle shopping histories dataset contains 197,324 records with 600 binary features representing a purchase (yes/no) of a product class for different shopping styles. Foursquare check-in location dataset contains 5,010 users' profiles having 446 binary features corresponding to the different regions and type of venue visits. All these visits are classified into thirty geosocial types. The hospital discharge database from Texas contains 67,330 inpatient records with 6170 binary features, classification focuses on the 100 most common medical procedures. The MNIST handwritten digits, UCI Adult census income data, Special subsets for parameter sensitivity analysis form additional datasets.

5.1 Reproducibility Specifications

To ensure full reproducibility of the reported results, the following experimental parameters are specified:

Target Model Architecture: For image data (CIFAR-10, CIFAR-100, MNIST), the target model consists of two convolutional layers (32 and 64 filters, kernel size 3×3) with ReLU activation, max pooling (2×2), and two fully connected layers (512 and c units) with dropout (0.5). For tabular data (Purchase-100, Hospital-100, Location-30, Adult), the target model is a multi-layer perceptron with two hidden layers (256 and 128 units) and ReLU activation.

Attack Model Architecture: The attack model is a neural network with a single hidden layer of 64 units, ReLU activation, and a sigmoid output unit. The input dimension varies by dataset: for c-class problems, the input is the c-dimensional prediction vector.

Optimizer: Stochastic gradient descent (SGD) with Nesterov momentum of 0.9.

Loss Function: Cross-entropy loss for target models; binary cross-entropy for attack models.

Learning Rate: 10^{-3} for target models, decayed by a factor of 0.1 after 50 epochs without validation loss improvement; 10^{-2} for attack models.

Batch Size: 64 for target models; 128 for attack models.

Epochs: Maximum of 200 epochs for target models with early stopping (patience of 10 epochs based on validation accuracy); 100 epochs for attack models.

Initialization: Xavier uniform initialization for all weights; zero initialization for biases.
Random Seed: 42 for all experiments.
Train/Test Split: 80/20 random split stratified by class.
Number of Independent Runs: 5 independent runs with different random seeds.

5.2 Query-Cost Calculation

The reported query budget is derived as follows:

$$\text{Total Queries} = (N_{\text{shadow}} \times R_{\text{model}} \times Q_{\text{record}}) + Q_{\text{attack}} \tag{1}$$

where:

- N_{shadow} = number of shadow models = 20
- R_{model} = records per model = 10,000
- Q_{record} = queries per record = 156 (average for Purchase-100 with 600 binary features)
- Q_{attack} = attack-model queries = 400,000 (20 attack models × 20,000 queries each)
- Additional API calls = 0 (no additional calls beyond those counted)

Substituting the values:

$$\text{Total Queries} = (20 \times 10{,}000 \times 156) + 400{,}000 = 31{,}200{,}000 + 400{,}000 = 31{,}600{,}000 \tag{2}$$

This yields approximately 156 queries per synthetic record and a total budget below 50 million queries, as reported in the manuscript.

The target models are implemented using three platforms: (a) Google prediction API which is fully automated in its configuration, (b) Amazon Machine Learning with default (10 passes, L2=1e-6) and regularized (100 passes, L2=1e-4) settings, and (c) neural networks locally trained in Torch7 with architectures specific to the dataset. We employ two convolution plus pooling layers and fully connected layers for image data with convolutional networks; a single hidden layer network with 128 hidden units suffices for tabular data. All models were trained using stochastic gradient descent with a batch size of 64, learning rate of 10^{-3} (decayed by a factor of 0.1 after 50 epochs without validation loss improvement), and a maximum of 200 epochs. Early stopping was applied with a patience of 10 epochs based on validation accuracy. Weight initialization used Xavier uniform initialization. These hyperparameters were held fixed across datasets to ensure consistent evaluation.

We adopt precision as the primary success indicator for membership inference attacks, because false positives (classifying a non-member as a member) carry different privacy consequences than false negatives in most regulatory contexts. Recall, F1-score, and accuracy are reported for completeness but are secondary to precision in our analysis.

Table 1 summarizes key experimental results. Google Prediction API models exhibit the highest vulnerability across most datasets, with purchase classification achieving 0.935 precision, 0.994 recall, and 0.964 F1-score—near-perfect membership detection. Amazon ML shows slightly reduced but still substantial vulnerability, while locally trained neural networks demonstrate intermediate leakage. The hospital discharge dataset, despite relatively low model accuracy (0.517 test accuracy), yields 0.657 attack

Table 1: Membership Inference Attack Performance

Data	Cls	Plat.	TrAcc	TeAcc	Prec	Rec	F1
Purchase	100	G-API	0.999	0.659	0.935	0.994	0.964
Purchase	100	AWS	0.992	0.683	0.840	0.988	0.908
Purchase	100	NN	0.997	0.672	0.870	1.000	0.930
Hospital	100	G-API	0.668	0.517	0.657	0.950	0.777
Location	30	G-API	1.000	0.673	0.678	0.980	0.801
CIFAR-10	10	NN	0.995	0.600	0.720	0.990	0.833
CIFAR-100	100	NN	0.998	0.200	0.980	0.995	0.987
Adult	2	G-API	0.848	0.842	0.503	0.510	0.506
MNIST	10	G-API	0.984	0.928	0.517	0.520	0.518

precision, confirming that healthcare data faces significant privacy risks. Notably, binary classification tasks (Adult dataset) and balanced image datasets (MNIST) show minimal vulnerability, with attack precision near random guessing at 0.503 and 0.517 respectively. The Adult dataset's resistance to membership inference warrants further explanation. Adult contains 48,842 records with 14 mixed-type features, and the target model achieves near-identical training and test accuracy (0.848 vs. 0.842), indicating minimal overfitting. Additionally, binary classifiers produce only a single probability value per prediction, limiting the feature space available to the attack model. The combination of low overfitting and low-dimensional prediction vectors makes membership patterns harder to distinguish. The results suggest a strong association between overfitting and membership-inference vulnerability. Overfitting serves as an important predictor of model susceptibility to privacy attacks, as the results indicate that models with higher degrees of overfitting tend to exhibit greater vulnerability. This association is consistent across all seven datasets evaluated in our study.

The relationship between model overfitting and attack success emerges clearly from our experiments. Models with large gaps between training and test accuracy indicating overfitting consistently show higher vulnerability. For instance, CIFAR-100 neural networks achieve near-perfect training accuracy (0.998) but poor generalization (0.200 test accuracy), resulting in extremely high attack precision (0.980). Conversely, Adult census models show minimal overfitting (0.848 train vs 0.842 test accuracy) and correspondingly low vulnerability (0.503 precision). This correlation confirms theoretical expectations that memorization of training data facilitates membership inference.

Class distribution imbalance significantly influences per-class attack accuracy. Minority classes with few training examples exhibit both higher overfitting and higher vulnerability, as models learn idiosyncratic patterns specific to those limited examples. Majority classes with abundant training data show better generalization and reduced leakage. Figure 2 illustrates this phenomenon across platforms, with Google API showing consistently higher precision due to more aggressive fitting to training data. The number of output classes also affects vulnerability: models with more classes (CIFAR-100 vs CIFAR-10, 100-class vs 10-class purchase models) leak more information, as they must memorize more discriminative features.

Shadow data generation method significantly impacts attack effectiveness but not feasibility. Model-based synthesis using only target API access achieves 0.896 precision on purchase data—only marginally below the 0.935 achieved with real shadow data. Statistics-based synthesis yields 0.795 precision, demonstrating that even approximate distribution knowledge enables effective attacks. Noisy real data with 10% feature randomization maintains 0.666 precision versus 0.678 with clean data, confirming robustness to imperfect distribution matching. These results establish that membership inference attacks remain potent across varying adversary knowledge levels.

Query efficiency analysis reveals practical attack feasibility. Model-based synthesis requires approximately 156 queries per synthetic record on average for purchase data with 600 binary features. With 20

shadow models each trained on 10,000 records, total query costs remain below 50 million—substantial but feasible for determined adversaries. Statistics-based approaches require no target queries during data generation, only during final attack execution. These query budgets fall within plausible ranges for external adversaries monitoring production ML APIs over extended periods.

Cross-platform vulnerability differences highlight the importance of training algorithm choices. Google's fully automated configuration produces models highly susceptible to membership inference, while Amazon's configurable regularization provides measurable though incomplete protection. Locally trained neural networks with carefully tuned regularization achieve intermediate vulnerability, suggesting that algorithm awareness alone cannot eliminate leakage without explicit privacy mechanisms. These findings underscore the need for platform-level privacy enhancements rather than relying on consumer expertise [13].

6. Vulnerability Factors and Mitigation Analysis

Our experimental analysis reveals several key factors influencing membership inference vulnerability, providing insights for both attack optimization and defense design. The most significant factor is model overfitting, quantified by the gap between training and test accuracy. Models that memorize training examples rather than learning generalizable patterns exhibit distinctive prediction behaviors on those memorized examples, primarily through higher confidence scores and more concentrated probability distributions. This behavioral signature provides the primary signal exploited by our attack methodology. Interestingly, overfitting alone does not fully explain vulnerability differences across platforms; model architecture and training algorithms introduce additional variation [5].

The number of output classes represents a second critical factor. Multi-class classification models with numerous output categories (50-100) demonstrate substantially higher vulnerability than binary or few-class models. This relationship stems from the increased discriminative information available in prediction vectors: with more classes, the probability distribution contains more dimensions where membership-related patterns can manifest. Additionally, multi-class models often employ more complex architectures with greater capacity for memorization. Binary classifiers like the Adult income predictor show minimal vulnerability despite reasonable accuracy, as prediction vectors contain essentially one degree of freedom (the probability of positive class).

Class distribution imbalance introduces heterogeneous vulnerability across a single model. Minority classes with limited training examples suffer disproportionate leakage, as models overfit more severely to small samples. Our experiments show attack precision varying by over 0.3 between well-represented and underrepresented classes in the same model. This finding has important implications for fairness and equity in privacy protection: individuals belonging to minority groups may face greater privacy risks when their data contributes to ML models. The relationship is non-monotonic: extremely rare classes (below 0.5% of training data) sometimes show reduced vulnerability because models fail to learn meaningful patterns, but this provides little comfort as prediction utility for these classes is also compromised.

Training dataset size exhibits complex relationships with vulnerability. For fixed model capacity, increasing training data generally reduces overfitting and thus vulnerability. However, with fixed per-class data, increasing the number of classes (and thus total data) can increase vulnerability due to the class count effect previously described. The most vulnerable configuration in our experiments combines moderate total data (10,000 records) with many classes (100), creating conditions where models memorize patterns without achieving true generalization. Very large datasets (50,000+ records) with balanced classes show

Table 2: Effectiveness of Mitigation Strategies (Purchase-100). Type: T = training-time defense, O = output-level defense. Acc.: test accuracy; Prec.: attack precision; Rec.: attack recall; F1: F1-score; Red.: reduction in attack precision relative to the "None" baseline, computed as $(0.935 - \text{Prec.})/0.935 \times 100\%$.

Strategy	**Type**	**Acc.**	**Prec.**	**Rec.**	**F1**	**Red.**
None	–	0.659	0.935	0.994	0.964	–
Top-3 only	O	0.659	0.870	0.990	0.926	7.0
Top-1 only	O	0.659	0.830	1.000	0.907	11.2
Label only	O	0.659	0.600	0.990	0.747	35.8
Round (3 dig.)	O	0.659	0.870	0.990	0.926	7.0
Round (1 dig.)	O	0.659	0.830	1.000	0.907	11.2
Temp. (t=5)	O	0.659	0.860	0.930	0.894	8.0
Temp. (t=20)	O	0.659	0.830	0.860	0.845	11.2
L2 (10^{-4})	T	0.680	0.810	0.960	0.879	13.4
L2 (10^{-3})	T	0.720	0.730	0.860	0.790	21.9
L2 (10^{-2})	T	0.630	0.540	0.520	0.530	42.2

reduced vulnerability, though still significantly above random guessing.

6.1 Mitigation Implementation Details

The following implementations are used for each mitigation strategy:

Top-k Output: This defense modifies only the returned prediction, not the model training. For a given query, we select the top-k class probabilities and renormalize them to sum to 1, discarding all other classes. For top-3 and top-1 variants, the probabilities are renormalized over only the retained classes. This does not affect the underlying model parameters.

Label-only Output: This defense modifies only the returned prediction by returning only the predicted class label (argmax), discarding all probability information. The model training remains unchanged. This provides maximum output truncation but loses all uncertainty information.

Temperature Scaling: This defense modifies only the returned prediction by applying a temperature parameter t to the logits before the softmax function. Specifically, for logits z_i, the output probabilities become $\text{softmax}(z_i/t)$. Higher temperatures produce more uniform probability distributions. This does not modify model training; it is applied as a post-processing step at inference time.

Regularization (L2): This defense modifies model training by adding an L2 weight decay penalty to the loss function: $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{original}} + \lambda\|\mathbf{w}\|_2^2$, where λ is the regularization strength. This directly constrains model complexity during training, reducing overfitting, and is a training-time defense.

Prediction vector granularity directly controls information leakage. Full probability vectors with floating-point precision provide maximum signal for membership inference. Truncating output to top-k classes or reducing numerical precision progressively reduces vulnerability, but with diminishing returns. As shown in Table 2, limiting output to only the predicted class label (no probabilities) reduces attack precision from 0.935 to 0.600, still above random guessing but substantially mitigated. This residual vulnerability stems from the attack model's ability to learn membership patterns from misclassification distributions: members and non-members are misclassified to different incorrect classes at different rates.

We evaluate several mitigation strategies for practical deployment. Output truncation approaches including top-k filtering and precision reduction provide partial protection with no accuracy cost, making them attractive for immediate implementation. However, they degrade model interpretability and may impact downstream applications using full probability distributions. Temperature scaling in softmax layers increases prediction entropy, reducing confidence differentials between members and non-members. This technique offers tunable privacy-utility tradeoffs but requires model architecture modifications not

always feasible in MLaaS settings.

Regularization techniques including L2 weight decay directly address overfitting, the root cause of vulnerability. As shown in Table 2, increasing regularization strength progressively reduces attack success, with $\lambda = 10^{-2}$ achieving 42.2% precision reduction. However, excessive regularization eventually harms test accuracy, creating utility tradeoffs. The optimal regularization level balances generalization improvement against accuracy preservation, typically reducing but not eliminating vulnerability. Platform providers could implement adaptive regularization based on privacy risk assessment of the training data.

Differential privacy (DP) provides theoretical guarantees against membership inference by design. However, practical DP implementations for complex models often incur substantial accuracy degradation or require non-trivial hyperparameter tuning. While not evaluated in our experiments due to limited MLaaS support, DP represents the gold standard for formal privacy protection. Future ML platforms should integrate DP training options with clear privacy-accuracy tradeoff explanations for consumers [14, 15].

Platform-level interventions offer the most promising path for widespread vulnerability reduction. MLaaS providers could implement privacy-aware model selection algorithms that consider not only accuracy but also inferred privacy risk based on dataset characteristics. Privacy risk scores could guide automatic regularization tuning or trigger warnings for high-risk configurations. Additionally, platforms could offer privacy-enhancing output filters as configurable API options, allowing consumers to select appropriate privacy-utility tradeoffs for their applications.

The regulatory implications of our findings are substantial. Current data protection frameworks like GDPR and HIPAA focus primarily on direct data access controls, with limited provisions for inference-based privacy violations. Our demonstration that production ML systems leak training data membership at high rates suggests regulatory gaps needing attention. Compliance assessments should include membership inference testing for ML models processing sensitive data, and privacy impact statements should quantify inference risks alongside traditional access risks [1, 13].

7. Conclusion

Membership Inference is a serious privacy threat in today's production machine learning systems especially those found in cloud-based MLaaS platforms. Through the formulation of the shadow model attack methodology and the thorough assessment over a diversified spectrum of real-world datasets, we exhibit that adversaries possessing merely black-box API access can achieve high accurate identification of whether a record was part of the private training dataset. Based on our experiments, the median attack precision on Google Prediction API was 0.657 for the hospital dataset, while the median attack precision on Amazon ML was 0.678 for the location dataset. Despite relatively low model accuracy, healthcare data shows vulnerability of approximately 65.7% precision.

The shadow model technique provides a unified evaluation methodology that enables direct comparison across datasets, the analysis of query budgets in realistic adversarial scenarios, and the characterization of defense effectiveness across different threat models. This framework provides actionable insights for deploying privacy-preserving machine learning in practice. Training proxy models on synthetically generated data to learn their behavior on known membership status and provide membership inference as a standard classification problem solvable by regular ML methods. The efficacy of the approach is retained in instances when the adversary possesses knowledge of either purely API access or an approximate knowledge of the distribution [5].

Our investigation uncovers crucial factors behind the model's insecurity. Among these, the excessive

fitting is of paramount significance. The results suggest a strong association between overfitting and membership-inference vulnerability, with overfitting serving as an important predictor of model susceptibility. While these factors correlate with attack success, it is hard to pin vulnerabilities down to just one of them, indicating the interplay of architectural, algorithmic, and data-related characteristics. The difficulty in designing universal defenses and the need for a holistic privacy assessment framework is highlighted by this [2, 4].

While several strategies appear only partially effective, no combination completely eliminates vulnerability without high utility tradeoffs. Truncating output and diminishing precision offer a quantifiable level of safeguarding that does not impact accuracy, however they do compromise interpretability of the model. Regularization tackles the root causes, but the test accuracy suffers in the end. The low accuracy guarantees of differential privacy can be hard to implement and expensive to use. Configurable output filters and privacy-aware model selection can be practically deployed at platform level [3].

There are serious regulatory impacts No concrete data protection framework deals with inference-based privacy violations, making compliance issues for organizations applying ML systems. The results of our study recommend that regulators expand their view to cope with indirect data leakage arising out of model behaviours. There must be standardized privacy testing methods to test ML systems. Moreover, there must be clarity of liabilities of platform providers and model consumers [13].

Future research should examine several avenues. We should develop improved mitigation techniques which efficiently and effectively balance privacy and utility, ideally through adaptive methods that take stronger action only if inference risk is high. Better detection techniques for ongoing membership inference attacks will enable defenders to identify and respond to privacy breaches. Formal privacy guarantees can offer solid foundations for secure ML deployment, given certain model architectures and training algorithms. The design of MLaaS platforms that maintain the privacy of their users such as differentially private MLaaS, secure multi-party computation or homomorphic encryption- which provide end-to-end privacy while being user-friendly would prove beneficial in the impending future.

With the growing adoption of machine learning in sensitive domains, the threat of membership inference is no longer an academic issue. The methodology, empirical evidence and analytical framework developed in this work can boost both attack and defence to enhance privacy-aware machine learning ecosystems. Our approach seeks to enrich the technical design and policy development for trustworthy AI systems by quantifying vulnerabilities and assessing countermeasures [1].

References

[1] Ayodeji Oseni, Nour Moustafa, Helge Janicke, Peng Liu, Zahir Tari, and Athanasios V. Vasilakos. Security and privacy for artificial intelligence: Opportunities and challenges. *arXiv preprint arXiv:2102.04661*, 2021.

[2] Muhammad Danish and Malik Muhammad Siraj. Ai and cybersecurity: Defending data and privacy in the digital age. *Journal of Engineering and Computational Intelligence Review*, 3(1):25–35, 2025.

[3] Joseph Nnaemeka Chukwunweike, Moshood Yussuf, Oluwatobiloba Okusi, Temitope Oluwatobi Bakare, and Ayokunle J. Abisola. The role of deep learning in ensuring privacy integrity and security: Applications in ai-driven cybersecurity solutions. *World Journal of Advanced Research and Reviews*, 23(2):1778–1790, 2024. doi: 10.30574/wjarr.2024.23.2.2550.

[4] Bhavani M. Thuraisingham. Can ai be for good in the midst of cyber attacks and privacy violations?:

A position paper. In *Proceedings of the Tenth ACM Conference on Data and Application Security and Privacy*, pages 1–4, 2020.

[5] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pages 506–519, 2017.

[6] Moustafa Alzantot, Yash Sharma, Supriyo Chakraborty, Huan Zhang, Cho-Jui Hsieh, and Mani B Srivastava. Genattack: Practical black-box attacks with gradient-free optimization. In *Proceedings of the genetic and evolutionary computation conference*, pages 1111–1119, 2019.

[7] Sultan Alneyadi, Elankayer Sithirasenan, and Vallipuram Muthukkumarasamy. A survey on data leakage prevention systems. *Journal of Network and Computer Applications*, 62:137–152, 2016.

[8] Suvendu Kumar Nayak and Ananta Charan Ojha. Data leakage detection and prevention: Review and research directions. *Machine learning and information processing: proceedings of ICMLIP 2019*, pages 203–212, 2020.

[9] Lanqing Guo, Chong Wang, Wenhan Yang, Siyu Huang, Yufei Wang, Hanspeter Pfister, and Bihan Wen. Shadowdiffusion: When degradation prior meets diffusion model for shadow removal. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14049–14058. IEEE, 2023.

[10] Krishna Sai Penumarthi. Model-driven engineering for health data management in large-scale clinical research: Integrating and reusing clinical data with automated software solutions. 2026. doi: 10.1109/GACLM67198.2025.11232132.

[11] Sameeruddin Shaik. Privacy and security in instant messaging: Evaluating telegram's cryptographic framework. In *2025 International Conference on Next Generation Computing Systems (ICNGCS)*, pages 1–12, 2025. doi: 10.1109/ICNGCS64900.2025.11183529.

[12] Vihar Kuruppathukattil. Secure data transmission in cloud environments: A layered approach. In *2025 IEEE 3rd Global Conference on Wireless Computing and Networking (GCWCN)*, pages 1–6, 2025. doi: 10.1109/GCWCN66157.2025.11448363.

[13] Prashanth Chevva. Balancing accuracy and efficiency in distributed machine learning systems: Regulatory implications and challenges. In *Proceedings of the International Conference*, 2025. URL `https://www.atlantis-press.com/proceedings/icaaai-25/126012569`.

[14] Taher M. Ghazal Tushita Agarwal, Nidal A. Al-Dmour. Privacy-preserving aggregation in iot sensor networks via federated averaging algorithms. *2025 International Conference on Computing and Radar (ICCR)*, 2025. doi: 10.1109/ICCR67387.2025.11292234.

[15] Shruti Hardia. Decentralized intelligence for smart cities: Integrating federated learning, blockchain, and foundation models for privacy-preserving urban data management. In *Proceedings of the 2025 International Symposium on Autonomous Decentralized Systems (ISADS)*, 2025. doi: 10.1109/ISADS66912.2025.00017. URL `https://doi.org/10.1109/ISADS66912.2025.00017`.

[16] Shantanu Sudhir Gujar. Exploring device fingerprinting for password-less authentication systems. In *2024 Global Conference on Communications and Information Technologies (GCCIT)*, 2024. doi: 10.1109/GCCIT63234.2024.10862062.

[17] Shiva Kumar Bommakanti. Cloud system observability: Automating infrastructure insight for service level adherence. In *2025 International Conference*, 2025. doi: 10.1109/ACOIT66109.2025.11436790.

[18] Akhil Veluru. Enhancing bibliometric insights: A novel overlapping clustering framework for citation networks. engrXiv preprint, 2026. URL https://doi.org/10.31224/5011.

[19] Pratik Dhameliya. Event-trace cohesion metrics for live co-authoring streams. 2026. doi: 10.1109/ICPCSN68523.2026.11543956.

[20] Apeksha Bhuekar. Latent phrase-aware generative modeling for expressive symbolic audio synthesis. 2026. doi: 10.20944/preprints202603.0308.v1.

[21] Tahmina Anondi. Performance dynamics in big data cloud streaming systems. 2025. doi: 10.1109/RAAI67517.2025.11423241. URL https://doi.org/10.1109/RAAI67517.2025.11423241.

[22] Bhuvan Chandra Sarakam. High-performance real-time data store. 2026. doi: 10.5281/zenodo.19709103. URL https://doi.org/10.5281/zenodo.19709103.

[23] Shujaatali Badami. Optimizing lsm tree operations with deferred updates: A comparative study. In *2024 International Conference on Data Science and Information Technology*, 2024. doi: 10.1109/DSIT61374.2024.10881309.

[24] Guruprasath Sankaran. Multi-crop recommendation using xgboost: A machine learning approach for sustainable agriculture. 2026. doi: 10.67231/23f69679. URL https://doi.org/10.67231/23f69679.

[25] Kaushal Thaker. Optimizing social media analytics with apache spark. doi: https://doi.org/10.31224/5114.

[26] Krutika Shah. Data tier architectures for enterprise saas: A comparative performance study of tenant isolation models. *AI/ML and Enterprise Systems*, 2026. doi: 10.67231/x7z7ys59. URL https://doi.org/10.67231/x7z7ys59.

[27] Uday Kumar Gannu. Tuple-indexed labeling: In-place structures for scalable graph metadata. In *Proceedings of the 4th IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)*, 2026. doi: 10.1109/IATMSI68868.2026.11466267.

[28] Suraj Davariya. Ai-powered detection of software code anomalies using deep learning. 2026. URL https://ieeexplore.ieee.org/document/11447954. Date of Conference: 12-13 December 2025, Bhopal, India.

[29] Krishna Kumar. Enhancing web accessibility for mathematics through semantic enrichment of mathml. 2025. doi: 10.1109/ICCTDC64446.2025.11158762.