From Black-box to Label-only: a Plug-and-Play Attack Network for Model Inversion

Huan Bao[1, 2, 3, 4]
lingyan@stu2022.jnu.edu.cn
Kaimin Wei*[1, 2, 3, 4]
cswei@jnu.edu.cn
Yao Chen[1, 2, 3, 4]
csyaochen@gmail.com
Hanting Hou[1, 2, 3, 4]
nofresh1043194045@gmail.com
Jinpeng Chen[5, 6]
jpchen@bupt.edu.cn
Yongdong Wu*[1, 2, 3, 4]
wuyd00@qq.com

[1] National Joint Engineering Research Center for Network Security Detection and Protection Technology

[2] Guangdong Key Laboratory for Data Security and Privacy Preserving

[3] Guangdong-Hong Kong Joint Laboratory for Data Security and Privacy Preserving

[4] College of Cyber Security, Jinan University
Guangzhou 510632, P. R. China

[5] School of Computer Science
Beijing University of Posts and Telecommunications
Beijing, 100876, P. R. China

[6] Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education
Jilin University
Changchun, 130012, P. R. China

Abstract

Model inversion (MI) attacks can reconstruct the private training data of deep neural networks. Nevertheless, existing black-box MI attacks significantly depend on the soft labels obtained from the classifier of the target model, which restricts their practicality. In this paper, we present a generic Plug & Play Attack Network (PnPAN) for MI, which is the first general framework to transform black-box MI attacks into hard-label-only ones. The fundamental idea of this framework is to assess the existing latent code exclusively using hard labels and employ a pre-trained reverse network. This might make it possible to reconstruct the classifier of the target model with just hard labels. Extensive experimental results demonstrate our approach's performance superiority compared with the state-of-the-art label-only attack and its broad applicability.

*Corresponding author

1 Introduction

In recent years, Deep Neural Networks (DNNs) have been increasingly prevalent in various domains [1, 2, 3, 4], particularly in handling personal and sensitive datasets such as medical records or financial transactions. Although the widespread use of DNN models has made things more convenient, it has also sparked worries about possible privacy breaches [5, 6, 7]. Model inversion (MI) attacks [7, 8, 9, 10, 11, 12, 13, 14, 15] are privacy attacks that try to uncover the original privacy training data of a model by understanding its behavior. These attacks are particularly destructive.

MI attacks heavily rely on the level of access the attacker has to the target model and the extent of their relevant previous knowledge. They can be classified into three forms based on several attack assumptions: white-box assault, black-box attack, and label-only attack. The white-box MI attack [9, 12, 16] grants the attacker unrestricted access to the target model, encompassing its internal architecture, parameter settings, and intermediate gradient data. In addition, the attacker may have access to privacy data that is intentionally made unclear, which can be utilized to assist in the process of reconstructing privacy. The black-box MI attack greatly limits the attacker's possibilities [7, 10, 11]. The only source of information available to them is the classifier of the target model, which provides confidence scores and labels. Furthermore, they are unable to retrieve any privacy-related data to assist with the rebuilding process. The label-only MI attack significantly restricts the attacker's capabilities [17]. They are only able to obtain the explicit classifications made by the target model, without any other data like confidence scores.

Black-box attacks demonstrate remarkably high levels of success in terms of both the rate of reconstruction and the quality of the results. Nevertheless, in real applications, DNNs usually produce only one definitive classification label when used for classification tasks. This constraint makes earlier assumptions about black-box attacks useless, therefore making label-only attacks more appropriate for such scenarios. Although BERP-MI [17] is the first label-only attack, it has numerous notable limitations. On the one hand, BERP-MI necessitates the identification of a latent code belonging to the target class in order to initiate the attack in the initialization phase. This can result in failure after a small number of queries or even indefinite initialization. On the other hand, BERP-MI merely provides an approximation of the optimization direction by utilizing hard labels, without completely capitalizing on the information contained in these labels. As a result, its total effectiveness is limited.

To end the aforementioned issues, we develop a generic framework named Plug & Play Attack Network (PnPAN). This framework exclusively depends on hard labels to reconstruct the classifier of the target model. This enables the reconstructed classifier to easily substitute the data employed in existing black-box attacks, thereby transforming them into attacks that solely focus on the labels. Specifically, we begin the process by sampling the nearby latent space around the current latent code. We then utilize the hard labels produced by the target model to assess the classification accuracy and prediction vector of the current latent code in relation to the target model. Next, we reconstruct the feature vector using a pre-trained reverse model. PnPAN replaces the existing classifier in the target model, enabling current black-box MI attacks to label-only attacks. To summarize, our contributions encompass:

- To the best of our knowledge, PnPAN is the first general framework that transforms black-box MI attacks into label-only ones.
- PnPAN eliminates unrealistic assumptions seen in current black-box MI attacks, making label-only attacks highly destructive and challenging to counteract.

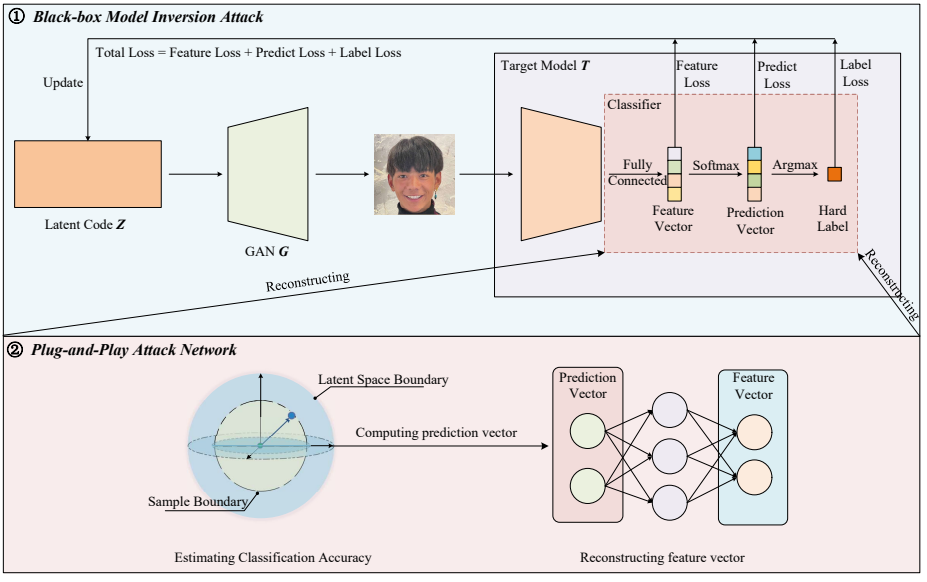


Figure 1: The structure of existing black-box MI attacks and the Plug & Play Attack Network (PnPAN). ① Existing black-box MI attacks rely on the soft labels obtained from the classifier of the target model. ② The PnPAN framework consists of two main components: a reverse network that reconstructs the feature vector, and an approximation portion that estimates the classification accuracy of the current latent coding using hard labels.

- We apply PnPAN to the state-of-the-art black-box MI attacks. Extensive experimental results demonstrate the generality of our method and highlight its superior reconstruction performance, surpassing even the most advanced label-only attacks available.

The rest of this paper is structured as follows. Sec. 2 presents related work and analyzes the current black-box attacks. Sec. 3 provides a detailed introduction to our approach. Sec. 4 discusses the experimental results. Finally, Sec. 5 summarizes this work.

2 Related Work

2.1 Model Inversion Attack

Early MI attacks. Fredrikson et al.[8] first proposed Model Inversion (MI) attacks against linear regression models for drug privacy prediction. Subsequently, Fredrikson et al.[18] developed confidence-based MI attack techniques capable of reconstructing grayscale facial images on simplified network structures. However, these early methods primarily targeted networks with fewer layers and were limited to grayscale image inputs, restricting their applicability. With the rapid advancement of deep learning technology, modern Deep Neural Networks (DNNs) exhibit more complex network architectures and higher data processing dimensions, challenging the applicability and effectiveness of traditional MI attack methods. There is an urgent need for in-depth research and improvement of these methods to adapt to DNNs.

GAN-based MI attacks. Research has gradually shifted towards MI attack techniques based on Generative Adversarial Networks (GANs) better to align MI attack methods with

real-world application scenarios. Zhang et al.[16] first proposed a white-box MI attack framework based on GANs. They trained GANs using public datasets to endow them with powerful image-generation capabilities. Furthermore, by optimizing the latent code of GANs, they generated privacy information closer to the target data, effectively reducing the cross-entropy loss of the target model on specific categories. Simplifying the optimization process by using latent code as the optimization target, they successfully implemented an MI attack on DNNs. Subsequently, Chen et al.[9] utilized the target model to generate pseudo-labels for public datasets, training GANs to endow them with label discrimination capabilities. Later, Yuan et al. [12] further improved GAN training strategies by selecting datasets and picking out data with high confidence for the target model on specific categories for training. This improved the representativeness of the training dataset and enabling the trained GAN to contain richer privacy information.

In white-box attacks, attackers may have access to model parameters or even acquire default or blurred versions of private data as prior knowledge. This scenario is highly unrealistic in real-world situations. Therefore, research on black-box MI attacks becomes essential. An et al.[11] proposed MIRROR utilized genetic algorithms instead of gradient information required in white-box settings, achieving high-precision reconstruction of private facial images from DNNs. Han et al.[10] employed reinforcement learning methods, modelling the search process of latent code as a Markov process and successfully executing MI attacks with only soft labels. On the other hand, Bao et al.[7] optimized latent distribution through multi-agent reinforcement learning and sample latent code from this distribution to achieve an MI attack. Additionally, Kahla et al. [17] introduced a label-only attack method by estimating the gradient of the current latent code using hard labels. The estimated gradient is then utilized to optimize the current latent code, achieving reconstruction of the target class.

2.2 Rethink Existing Black-box MI Attacks

The assumption of existing black-box MI attacks. Black-box MI attacks typically assume the existence of a target model T trained on a private face recognition dataset D_{priv}, capable of classifying K categories. Unlike the real-world scenario where only labels can be obtained, existing black-box MI attacks assume that attackers can directly access the classifier of the target model T. Expressly, the model accepts input x, processes it through fully connected layers to produce a feature vector $F_v = [f_{v1}, f_{v2}, ..., f_{vn}]$, and then maps it to a prediction vector $P_v = [p_{v1}, p_{v2}, ..., p_{vn}]$ via a softmax layer. Based on this, attackers compute the maximum value in the prediction vector $\max_i p_i = p_{\max}$ and determine its corresponding index $\hat{l} = \arg\max_i p_i$ as the output label of x relative to the model. At the same time, they obtain the classification accuracy $\max_i p_i$ of the model for x.

In addition, the black-box GAN-based MI attack also assumes that the attacker possesses a publicly available dataset D_{pub} of the same type as the private target dataset D_{priv}. Here, the same type refers to both being face recognition datasets, but they do not contain the same individual categories. The public dataset D_{pub} is used to train the GAN G as prior knowledge for the attacker.

The steps of existing black-box MI attacks. For a GAN-based MI attack, the attack objective is to construct an appropriate latent code z to generate suitable image data $x = G(z)$ using GAN G, revealing the private data of the target label y in D_{priv}. During the attack process, the attacker utilizes the obtained feature vector F_v, prediction vector P_v, predicted label $\hat{l}$, and the corresponding classification accuracy $\max_i p_i$ to respectively calculate the feature loss $L_{\text{feat}}()$, label loss $L_{\text{label}}()$, and prediction loss $L_{\text{pre}}()$ to construct the optimization

objective, formalized as:

$$argmin_x(L_{feat}(x;y)+L_{pre}(x;y)+L_{label}(\max_i p_i)) \tag{1}$$

Through optimization, the attacker ultimately reconstructs the data x that best represents the target privacy, revealing the private information of the target label y. The detailed steps of black-box MI attacks are shown in Fig.1.

3 Methodology: PnPAN

3.1 Threat Model

Target model. The target model T, when given input x, only outputs the hard label $y = T(x)$, and does not provide any other information beyond this.

Attacker knowledge. We assume that attackers can infer the function of the model based on its inputs and outputs. Meanwhile, attackers also can access publicly available datasets relevant to the model. In addition, the public and private datasets do not share any information. Attackers do not know the architecture of the model.

Attack goal. Attackers aim to rebuild the specific labels y^* of the model training data x^* by accessing this model and utilizing its hard labels.

3.2 Plug-and-Play Attack Network (PnPAN)

Fig.1 illustrates the structure of the Plug-and-Play Attack Network. In PnPAN, the surrounding latent space around the current latent code is sampled, and its classification accuracy is estimated by using the hard labels outputted by the model. Meanwhile, the prediction vector is created, and a pre-trained reverse network is utilized to reconstruct the feature vector, ultimately rebuilding the target model's classifier.

Estimating classification accuracy. When discussing methods to approximate classification accuracy, our first consideration is the continuity of the latent space of GANs. We also evaluate the generalization ability and decision boundaries of the target model. The continuity of the latent space of GANs means that small changes in the latent space will result in minor modifications to the generated images. A well-trained GAN should contain certain privacy information, allowing us to extract this information from the latent space to reconstruct private data. At the same time, the target model should have clear decision boundaries to distinguish between different categories of images effectively. Based on these considerations, we propose to set a sampling radius r and sample c times within the region where the current latent encoding z resides, obtaining a set of latent codes $Z = \{z_1, z_2, ..., z_c\}$ as follows, we can calculate one of latent code z_i as:

$$\mathbf{z_i} = \mathbf{z} + r \cdot \frac{\sigma}{||\sigma||} \tag{2}$$

Then, we utilize the GAN G to generate images and use the target model T for discrimination, obtaining a set of predicted labels $L = \{l_1, l_2, ..., l_n\}$. Given the target label l and one of predicted label $\hat{l}$, we define an indicator function $\mathrm{II}(\cdot)$ as:

$$II(l \cdot \hat{l}) = \begin{cases} 1, & if\ l = \hat{l} \\ 0, & otherwise \end{cases} \tag{3}$$

The classification accuracy p of the generated images is computed with the specific formula as follows:

$$p = \frac{\sum_{i=1}^{N} II(l \cdot l_i)}{N} \tag{4}$$

To achieve a substitute for the classification accuracy provided by the actual model.

Computing prediction vector. Next, we discussed how to compute the prediction vector. The model generates an n-dimensional feature vector $F_v = \{f_{v1}, f_{v2}, ..., f_{vn}\}$ by performing convolution and other operations on the images. Then, the prediction vector $P_v = \{p_{v1}, p_{v2}, ..., p_{vn}\}$ is calculated based on the following feature vectors.

$$p_{vi} = \frac{e^{f_{vi}}}{\sum_{j=1}^{n} e^{f_{vj}}} \tag{5}$$

Each p_{vi} in the prediction vector represents the classification accuracy for the corresponding label, and the sum of all p_{vi} is 1, we can construct the prediction vector to ensure that each p_{vi} belongs to the interval $[0,1]$. We can rewrite Eqn. 5 to construct the prediction vector:

$$p_{vi} = \frac{\sum_{j=1}^{N} II(i \cdot l_j)}{N} \tag{6}$$

However, when the sampling quantity N is less than the total number of classes K, some samples with low probabilities in real predictions may not be covered, leading to situations where specific probabilities are zero. To avoid this, we introduce an additional set of random noise $\sigma = \{\sigma_1, \sigma_2, ..., \sigma_n\}$, and the constructed prediction vector becomes:

$$p_{vi} = \alpha \frac{e^{f_{vi}}}{\sum_{j=1}^{n} e^{f_{vj}}} + (1-\alpha)\sigma_i, \ \sum_{i=1}^{N} \sigma_i = 1 \tag{7}$$

Through this method, we obtain an approximate prediction vector P_v, where each p_{vi} lies within the interval $(0,1]$, and the sum of all p_{vi} is 1.

Reconstruct feature vector. To solve for the feature vector, using Eqn. 6, we can obtain:

$$f_{vi} = \log(p_{vi}) + \log\left(\sum_{i=1}^{N} e^{f_{vi}}\right) \tag{8}$$

To reconstruct solve f_{vi}, we simplify its solution to:

$$f_{vi} = \mathbf{W^T} \log(p_{vi}) + C \tag{9}$$

where W^T and C are constants. This equation serves as the forward propagation function of the model during training. The loss is then defined as the mean squared error loss:

$$L = \frac{1}{N} \sum_{i=1}^{N} (f_{vi} - \hat{f}_{vi}) \tag{10}$$

Finally, we utilize the trained reverse model M to reconstruct the solution for the feature vector as follows:

$$f_{vi} = M(p_{vi}) \tag{11}$$

In conclusion, we can substitute information from the classifier in the target model by solely using hard labels.

4 Experiment

4.1 Experimental Setup

Datasets. In our experiments, we employed three datasets: CelebA [19], FaceScurb [20], and Pubfig83 [21]. Similar to previous studies, we constructed corresponding private and public datasets. The private dataset was employed to train the target model, while the public dataset was utilized to train the GAN. These two datasets were strictly separated to ensure no identity or data overlap. In addition, we also utilized FFHQ [22] as a public dataset to verify the impact of different data distributions on the MI attack performance.

Target models. We trained three popular deep face recognition networks, including FaceNet64 [23], ResNet152 [24], and VGG16 [25], as target networks using our private training dataset. To ensure the accuracy of performance evaluation, we also trained FaceNet, which has a more robust performance, using the same private training dataset as the evaluation model to assess the reconstruction results. This approach helps eliminate the potential impact of overfitting during the reconstruction process on performance evaluation, thus ensuring the reliability and accuracy of the evaluation results.

Baselines. We validated the effectiveness of our method by selecting some of the most advanced black-box MI attack methods available. Specifically, we chose the Reinforcement Learning-based Black-box Model Inversion Attacks (RLB-MI) [10] and the Distributional Black-Box Model Inversion Attack (DBB-MI) [7], adapting them into label-only MI attacks as our comparison baseline. We selected the Boundary-Repelling Model Inversion Attacks (BERP-MI) [17] as the comparative baseline for the label-only attacks to evaluate the performance under label-only conditions.

Evaluation metrics. To evaluate the performance of MI attacks, we assess whether the reconstructed images leak sensitive identity information related to the target labels. We employ both visual assessment and more objective quantitative methods to evaluate the effectiveness of the attack. The quantitative evaluation utilizes a range of metrics to assess attack performance, including attack accuracy, the proportion used to evaluate successfully reconstructed images, and KNN Dist and PSNR metrics for determining the quality of reconstructed images.

Hyperparameters. Through fine-tuning and experimentation, we select a sampling radius of 3 and 200 sampling iterations. For latent code z, we randomly select from the normal distribution. For the Reverse Model, we utilized VGG16, extracting feature vectors and prediction vectors from 20,000 randomly selected images from FFHQ for training. The settings for the label-only baseline and all black-box MI remain unchanged from their original configurations.

4.2 Experimental Results

Performance evaluation on different target models. Tab. 1 showcases the performance of our state-of-the-art black-box MI methods transformed into label-only MI methods against different target model architectures. Even with significantly reduced access to the model, we maintained a decent performance, with ACC decreasing by only 3.12%

Additionally, Fig. 2 illustrates the images reconstructed by employing PnPA for label-only MI attack. The reconstructed details and colours closely match or even surpass those of the current state-of-the-art black-box attacks, indicating the minimal impact of our method on the reconstruction outcome.

Figure 2: A comparison of the original images and the images reconstructed by various MI attacks. The top section depicts the original images from a private dataset. The middle part illustrates the images recreated by black-box MI attacks. The bottom part displays the images rebuilt by label-only MI attacks. The left column displays the names of the attacks. Each column represents an identical identity.

Model	Type	Method	ACC↑	PSNR↑	KNN Dist↓
VGG16	Black-box	RLB-MI	0.642	15.80	1262.34
		DBB-MI	0.858	20.66	1180.63
	Label-only	BERP-MI	0.562	13.36	1872.48
		RLB-MI+PnPA	0.490	15.31	1421.46
		DBB-MI+PnPA	**0.830**	**18.74**	**1253.15**
FaceNet64	Black-box	RLB-MI	0.804	16.27	1354.86
		DBB-MI	0.916	18.06	1091.15
	Label-only	BERP-MI	0.734	13.69	1685.29
		RLB-MI+PnPA	0.650	15.42	1398.40
		DBB-MI+PnPA	**0.900**	**16.63**	**1106.87**
ResNet-152	Black-box	RLB-MI	0.812	15.23	1308.69
		DBB-MI	0.898	17.37	1063.38
	Label-only	BERP-MI	0.754	13.17	1745.73
		RLB-MI+PnPA	0.660	14.91	1384.12
		DBB-MI+PnPA	**0.870**	**16.52**	**1165.72**

Table 1: The experimental results obtained by training different architecture models as target models on the CelebA dataset. In this table, symbols ↑ and ↓ indicate better performance in higher and lower metrics, respectively. The best results are obtained by label-only MI attacks and are highlighted in bold.

Performance evaluation on different dataset. Tab. 2 presents the experimental results obtained by training target models and GANs using various dataset partitions into private and public datasets. The improved black-box MI attack with PnPA surpasses the previous best label-only MI attack. On the CelebA dataset, DBB-MI with PnPA shows a 22.62% increase in ACC compared to BERP-MI, while on the FaceScurb dataset, RLB-MI with PnPA exhibits a 16.39% increase in ACC compared to BERP-MI. Performance degradation due to using PnPA is relatively lower on the FaceScurb and Pubfig83 datasets, as these datasets have fewer identities, resulting in a more accurate reconstruction of the classifier.

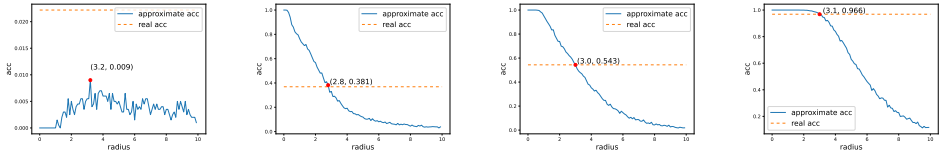
Performance evaluation on cross-dataset. The public and private datasets were sourced from the same dataset in previous experiments. Although there was no identity overlap between the two datasets, thereby avoiding direct privacy leakage, their similar collection and processing methods from the same dataset make their lighting resolution and other information relatively close, potentially leading to privacy information leakage. Moreover, in real-world scenarios, the training dataset of the target model is often unknown and obtaining a dataset with the same distribution as the target model is even more challenging than prior knowledge for the attacker. Therefore, we chose FFHQ as the public dataset for the attack GAN training. At the same time, the target models were trained using CelebA, FaceScurb, and Pubfig83 datasets, respectively, to ensure different dataset distributions, making the attack more practical. Tab. 3 demonstrates the MI attack results obtained under this setup. The DBB-MI improved with PnPA consistently outperformed the current state-of-the-art label-only MI attack on all datasets.

Table 2: The performance of MI attacks on different private datasets.

Dataset	Type	Method	ACC↑	PSNR↑	KNN Dist↓
CelebA	Black-box	RLB-MI	0.804	16.27	1354.86
		DBB-MI	0.916	18.06	1091.15
	Label-only	BERP-MI	0.734	13.69	1685.29
		RLB-MI+PnPA	0.650	15.42	1398.40
		DBB-MI+PnPA	**0.900**	**17.63**	**1106.87**
FaceScurb	Black-box	RLB-MI	0.420	19.23	2693.85
		DBB-MI	0.375	23.03	2661.82
	Label-only	BERP-MI	0.305	20.81	2684.91
		RLB-MI+PnPA	**0.350**	18.59	2713.20
		DBB-MI+PnPA	0.325	**21.51**	**2681.02**
Pubfig83	Black-box	RLB-MI	0.400	16.37	2349.84
		DBB-MI	0.560	17.04	2342.47
	Label-only	BERP-MI	0.400	13.10	2492.99
		RLB-MI+PnPA	0.340	15.42	2391.28
		DBB-MI+PnPA	**0.500**	**16.88**	**2331.96**

Table 3: The performance of MI attacks using the GAN trained on FFHQ.

Dataset	Type	Method	ACC↑	PSNR↑	KNN Dist↓
FFHQ ↓ CelebA	Black-box	RLB-MI	0.402	16.76	925.61
		DBB-MI	0.532	16.04	1002.51
	Label-only	BERP-MI	0.398	15.01	1331.25
		RLB-MI+PnPA	0.312	15.36	1342.86
		DBB-MI+PnPA	**0.456**	**15.96**	**1013.84**
FFHQ ↓ FaceScurb	Black-box	RLB-MI	0.355	16.56	2866.43
		DBB-MI	0.490	17.04	2817.85
	Label-only	BERP-MI	0.295	15.97	**2859.64**
		RLB-MI+PnPA	0.295	15.10	2901.63
		DBB-MI+PnPA	**0.425**	**16.89**	2873.57
FFHQ ↓ Pubfig83	Black-box	RLB-MI	0.360	16.05	2402.50
		DBB-MI	0.700	16.55	2387.56
	Label-only	BERP-MI	0.380	15.36	**2391.67**
		RLB-MI+PnPA	0.300	15.59	2452.96
		DBB-MI+PnPA	**0.66**	**16.07**	2403.67



(a) real acc = 0.020 (b) real acc = 0.381 (c) real acc = 0.543 (d) real acc = 0.966

Figure 3: A comparison of the real and the estimation accuracy values at different sampling radiuses. The dashed line represents the practical accuracy, while the other indicates the sampling radius closest to the practical accuracy.

In summary, by utilizing PnPA, we successfully enhanced the state-of-the-art black-box MI attack into a label-only one. It demonstrates robust performance across target models, datasets, and scenarios with more realistic distributions. Moreover, it surpasses the current state-of-the-art label-only MI attack in various aspects such as ACC, PSNR, KNN Dist, and visual quality evaluation. This indicates the effectiveness and versatility of PnPA in transforming black-box MI attack to label-only MI attack, allowing the current black-box MI attack to rely solely on the output labels of the target model while maintaining high performance.

4.3 Analysis factors affecting performance

The impact of sampling radius under different real accuracy. We first use the target model for a specific latent code to determine its accuracy. Then, using this latent code as the centre, we conducted 200 samplings at different radii to estimate its accuracy, aiming to verify the impact of sampling radius on estimation accuracy at different practical accuracy, as shown in Fig. 3. At a practical accuracy of 0.02, as shown in Fig. 3(a), the estimation accuracy exhibits fluctuations due to added noise, but this does not affect the overall optimization process. Oversized sampling radii may exceed the range of the target latent space, resulting in estimation accuracy that is far lower than the actual accuracy. Conversely, undersized sampling radii may result in most sampled values sharing the same label as the current latent code, leading to estimation accuracy far higher than the actual accuracy. Therefore, excessively high and low sampling radii can result in inaccurate estimation, affecting attack performance. In our experiments, the most accurate estimations were observed to fluctuate

Extract->Target	ACC↑	PSNR↑	KNN Dist↓
VGG16->VGG16	**0.830**	**18.74**	**1253.15**
FaceNet64->VGG16	0.804	17.89	1321.84

Table 4: The results obtained by training the reverse model with feature and prediction vectors extracted from different models.

around a radius of 3, which helps maintain the performance of the model inversion attack.

The impact of training data on the reverse model. We explored using datasets labelled with models of different structures to extract feature vectors and prediction vectors, which were then used to train reverse models for attacks. The results are shown in the Tab. 4. We used VGG16 and FaceNet64 to extract data from the same randomly selected FFHQ dataset to construct training data for the reverse model. We then inserted this reverse model into the DBB-MI attack on the VGG16 model for experimentation. The results are presented in the Tab. 4. Interestingly, the outcomes achieved with various reverse models were very similar. There was only a slight 3.23% decrease in accuracy observed when FaceNet64 was employed as the extraction model, in contrast to using a model that aligns with the target model. Hence, the training of the reverse model is independent of whether the extraction model matches the target model. This suggests that our framework can be deployed in scenarios where the structures of the target models are unknown.

5 Conclusion

In this paper, we reassess existing black-box model inversion techniques and propose a generic Plug & Play Attack Network (PnPAN) for MI attacks. It transforms the black-box MI attack, which relies on the target model classifier, into a label-only attack that requires hard labels. In PnPAN, the method involves sampling the latent space near the current latent code to estimate its prediction vector in relation to the target model. Additionally, a pre-trained inverse model is utilized to reconstruct the feature vectors of the target model. This allows us to reconstruct information from the classifier using only hard-label information. Extensive experimental results demonstrate that PnPAN can effectively convert black-box MI attacks into label-only MI attacks. Moreover, PnPAN achieves a 47.69% increase in accuracy compared to the state-of-the-art label-only attacks. This work highlights the substantial danger presented by Model Inversion Attacks and emphasizes the crucial significance of focusing on machine learning security.

ACKNOWLEDGEMENT

This work was in part supported by the National Natural Science Foundation of China (61932011), Guangdong Basic and Applied Basic Research Foundation (2019B1515120010), the Fundamental Research Funds for the Central Universities (21623402), the Science and Technology Program of Guangzhou (2024A04j6317), the Open Project of Anhui Provincial Key Laboratory of Multimodal Cognitive Computation, Anhui University (MMC202408), the Fundamental Research Funds for the Central Universities, JLU (93K172024K17), and the Open Project of Xiangjiang Laboratory (23XJ03006).

References

[1] Su-in Yi, Jack D Kendall, R Stanley Williams, and Suhas Kumar. Activity-difference training of deep neural networks using memristor crossbars. *Nature Electronics*,

6(1):45–51, 2023.

[2] Dimitrios Kollias, Anastasios Arsenos, and Stefanos Kollias. A deep neural architecture for harmonizing 3-d input data analysis and decision making in medical imaging. *Neurocomputing*, 542:126244, 2023.

[3] Ankit Thakkar and Ritika Lohiya. Fusion of statistical importance for feature selection in deep neural network-based intrusion detection system. *Information Fusion*, 90:353–363, 2023.

[4] Yasin Bayzidi, Alen Smajic, Jan David Schneider, Fabian Hüger, Ruby Moritz, and Alois C. Knoll. Performance limiting factors of deep neural networks for pedestrian detection. In *Proceedings of the 33rd British Machine Vision Conference (BMVC) 2022*, page 883, 2022.

[5] Dongliang Cao, Kaimin Wei, Yongdong Wu, Jilian Zhang, Bingwen Feng, and Jinpeng Chen. Fepn: A robust feature purification network to defend against adversarial examples. *Computers Security*, 134:103427, 2023.

[6] Shouhong Tan, Fengrui Hao, Tianlong Gu, Long Li, and Ming Liu. Collusive model poisoning attack in decentralized federated learning. *IEEE Transactions on Industrial Informatics*, 2023.

[7] Huan Bao, Kaimin Wei, Yongdong Wu, Jin Qian, and Robert H Deng. Distributional black-box model inversion attack with multi-agent reinforcement learning. *arXiv preprint arXiv:2404.13860*, 2024.

[8] Matthew Fredrikson, Eric Lantz, Somesh Jha, Simon Lin, David Page, and Thomas Ristenpart. Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing. In *Proceedings of the 23rd USENIX security symposium (USENIX Security 14)*, pages 17–32, 2014.

[9] Si Chen, Mostafa Kahla, Ruoxi Jia, and Guo-Jun Qi. Knowledge-enriched distributional model inversion attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16178–16187, 2021.

[10] Gyojin Han, Jaehyun Choi, Haeil Lee, and Junmo Kim. Reinforcement learning-based black-box model inversion attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20504–20513, 2023.

[11] Shengwei An, Guanhong Tao, Qiuling Xu, Yingqi Liu, Guangyu Shen, Yuan Yao, Jingwei Xu, and Xiangyu Zhang. Mirror: Model inversion for deep learning network with high fidelity. In *Proceedings of the 29th Network and Distributed System Security Symposium (NDSS)*, 2022.

[12] Xiaojian Yuan, Kejiang Chen, Jie Zhang, Weiming Zhang, Nenghai Yu, and Yang Zhang. Pseudo label-guided model inversion attack via conditional generative adversarial network. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 37, pages 3349–3357, 2023.

[13] Tianqing Zhu, Dayong Ye, Shuai Zhou, Bo Liu, and Wanlei Zhou. Label-only model inversion attacks: Attack with the least information. *IEEE Transactions on Information Forensics and Security*, 18:991–1005, 2022.

[14] Shuai Zhou, Tianqing Zhu, Dayong Ye, Xin Yu, and Wanlei Zhou. Boosting model inversion attacks with adversarial examples. *IEEE Transactions on Dependable and Secure Computing*, 2023.

[15] Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. Inverting gradients-how easy is it to break privacy in federated learning? *Advances in Neural Information Processing Systems (NIPS)*, 33:16937–16947, 2020.

[16] Yuheng Zhang, Ruoxi Jia, Hengzhi Pei, Wenxiao Wang, Bo Li, and Dawn Song. The secret revealer: Generative model-inversion attacks against deep neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 253–261, 2020.

[17] Mostafa Kahla, Si Chen, Hoang Anh Just, and Ruoxi Jia. Label-only model inversion attacks via boundary repulsion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15045–15053, 2022.

[18] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 1322–1333, 2015.

[19] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 3730–3738, 2015.

[20] Hong-Wei Ng and Stefan Winkler. A data-driven approach to cleaning large face datasets. In *Proceedings of the 2014 IEEE International Conference on Image Processing (ICIP)*, pages 343–347, 2014.

[21] Nicolas Pinto, Zak Stone, Todd Zickler, and David Cox. Scaling up biologically-inspired computer vision: A case study in unconstrained face recognition on facebook. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 35–42. IEEE, 2011.

[22] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4401–4410, 2019.

[23] Yu Cheng, Jian Zhao, Zhecan Wang, Yan Xu, Karlekar Jayashree, Shengmei Shen, and Jiashi Feng. Know you at one glance: A compact vector representation for low-shot learning. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops*, pages 1924–1932, 2017.

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[25] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2015.