

# Data Centric Defenses for Privacy Attacks

Nikhil Abhyankar

Thesis submitted to the Faculty of the  
Virginia Polytechnic Institute and State University  
in partial fulfillment of the requirements for the degree of

Master of Science  
in  
Computer Engineering

Ruoxi Jia, Chair  
Lynn Abbott  
Naren Ramakrishnan

July 20, 2023  
Blacksburg, Virginia

Keywords: Data Augmentation, Model Inversion, Membership Inference, Data Privacy

Copyright 2023, Nikhil Abhyankar

# Data Centric Defenses for Privacy Attacks

Nikhil Abhyankar

(ABSTRACT)

Recent research shows that machine learning algorithms are highly susceptible to attacks trying to extract sensitive information about the data used in model training. These attacks called privacy attacks, exploit the model training process. Contemporary defense techniques make alterations to the training algorithm. Such defenses are computationally expensive, cause a noticeable privacy-utility tradeoff, and require control over the training process. This thesis presents a data-centric approach using data augmentations to mitigate privacy attacks. We present privacy-focused data augmentations to change the sensitive data submitted to the model trainer. Compared to traditional defenses, our method provides more control to the individual data owner to protect one's private data. The defense is model-agnostic and does not require the data owner to have any sort of control over the model training. Privacy-preserving augmentations are implemented for two attacks namely membership inference and model inversion using two distinct techniques. While the proposed augmentations offer a better privacy-utility tradeoff on CIFAR-10 for membership inference, they reduce the reconstruction rate to  $\leq 1\%$  while reducing the classification accuracy by only 2% against model inversion attacks. This is the first attempt to defend model inversion and membership inference attacks using decentralized privacy protection.

# Data Centric Defenses for Privacy Attacks

Nikhil Abhyankar

(GENERAL AUDIENCE ABSTRACT)

Privacy attacks are threats posed to extract sensitive information about the data used to train machine learning models. As machine learning is used extensively for many applications, they have access to private information like financial records, medical history, etc depending on the application. It has been observed that machine learning models can leak the information they contain. As models tend to 'memorize' training data to some extent, even removing the data from the training set cannot prevent privacy leakage. As a result, the research community has focused its attention on developing defense techniques to prevent this information leakage. However, the existing defenses rely heavily on making alterations to the way a machine learning model is trained. This approach is termed as a model-centric approach wherein the model owner is responsible to make changes to the model algorithm to preserve data privacy. By doing this, the model performance is degraded while upholding data privacy. Our work introduces the first data-centric defense which provides the tools to protect the data to the data owner. We demonstrate the effectiveness of the proposed defense in providing protection while ensuring that the model performance is maintained to a great extent.

# Dedication

*To my grandparents, parents, friends, and teachers*



# Acknowledgments

I want to take a moment to express my sincere gratitude to my advisor Dr. Ruoxi Jia. Her guidance, encouragement, and patience have been pivotal in the completion of my project. Thank you Dr. Jia for your support throughout my Master's studies. I immensely value the time we spent trying to formulate novel ideas and the timely inputs you gave. I believe that this experience would help me as I take on future research projects.

I want to thank my committee members Dr. Lynn Abbott and Dr. Naren Ramakrishnan for providing me with timely suggestions and being a part of my advisory committee. I would like to express sincere gratitude to Dr. Mary Lanzerotti who has been a mentor and guide throughout my Master's degree. I wish to appreciate the members of ReDS Lab, Yi Zeng, Myeongseob Ko, Hoang Just, Himanshu Jahagirdar, Feiyang Kang, Minzhou Pan, and Si Chen.

I thank the Borgaonkar family, my sister Nikita, and my parents for their constant support. I thank my close friends Dishang, Arjun, Gaurang, Sanjana, Poonam, Pallavi, and Radhika for making my Master's memorable. Lastly, I want to thank Namita who has stood behind me like a rock through all the ups and downs.

# Contents

- List of Figures** **ix**
  
- List of Tables** **xi**
  
- 1 Introduction** **1**
  - 1.1 Motivation . . . . . 1
  - 1.2 Thesis Contributions . . . . . 6
  - 1.3 Thesis Outline . . . . . 7
  
- 2 Review of Literature** **8**
  - 2.1 Machine Learning . . . . . 8
  - 2.2 Privacy Attacks . . . . . 8
  - 2.3 Model Inversion . . . . . 9
    - 2.3.1 Model Inversion Attacks . . . . . 10
    - 2.3.2 Model Inversion Defenses . . . . . 11
  - 2.4 Membership Inference . . . . . 12
    - 2.4.1 Membership Inference Attacks . . . . . 13
    - 2.4.2 Membership Inference Defenses . . . . . 16
  - 2.5 Relation between Data Augmentation and Privacy . . . . . 19

<b>3</b>	<b>Model Inversion</b>	<b>20</b>
3.1	Proposed Methodology . . . . .	20
3.2	Experiments . . . . .	24
3.2.1	Attack Algorithm . . . . .	24
3.2.2	Evaluation Setup - Attacks, Models, and Datasets . . . . .	24
3.2.3	Comparison with Baseline Defenses . . . . .	25
3.3	Results . . . . .	26
3.4	Discussion . . . . .	30
3.4.1	Similarity between Target samples and Surrogate samples . . . . .	30
3.4.2	Non-zero diversity among surrogate samples within same class . . . . .	32
<b>4</b>	<b>Membership Inference</b>	<b>34</b>
4.1	Proposed Methodology . . . . .	34
4.2	Experiments . . . . .	38
4.2.1	Attack algorithm . . . . .	38
4.2.2	Baseline Augmentations for Comparison . . . . .	39
4.2.3	Evaluation Setup - Attacks, Datasets, and Models . . . . .	40
4.3	Results and Discussions . . . . .	41
<b>5</b>	<b>Conclusion and Future Work</b>	<b>47</b>
	<b>Bibliography</b>	<b>49</b>

<b>Appendices</b>	<b>57</b>
<b>Appendix A Datasets and Models</b>	<b>58</b>
A.1 Datasets . . . . .	58
A.2 Models . . . . .	60
<b>Appendix B Pseudo-Code of the proposed algorithms</b>	<b>62</b>
B.1 Model Inversion . . . . .	62
B.2 Memebership Inference . . . . .	63
<b>Appendix C Hyperparamters</b>	<b>64</b>
C.1 Model Inversion . . . . .	64
C.2 Membership Inference . . . . .	65

# List of Figures

1.1	Overview of White-Box attacks and Black-Box attacks . . . . .	3
1.2	Data Centric Defense vs Model Centric Defense . . . . .	5
2.1	Illustration of a model training algorithm and a model inversion attacks . . . . .	9
2.2	An example of the image recovered using model inversion attack. Compared with a sample image belonging to the target class, the recovered image shows similarity in the facial features . . . . .	10
2.3	Overview of a Membership Inference Attack. A set of data samples are passed through a trained ML model. The output is then given as an input to the membership inference attack which distinguishes between the training samples and the non-training samples. . . . .	13
3.1	Illustration of curvature-controlled augmentations and the resulting loss landscape of target and surrogate samples. . . . .	22
3.2	Visual representation comparing the faces recovered for different baselines by the Model Inversion Attack. Each row shows the reconstruction of an image belonging to the same identity. The true image is on the left followed by the reconstruction under no protection, DP-SGD, and MID . . . . .	27
3.3	Visual representation of recovered faces of Model Inversion Attacks . . . . .	31
4.1	Clean and augmented version of images . . . . .	36

4.2	Pipeline to generate augmented samples( $x + \delta$ ) for a target class from a pre-trained model $f_{\theta_{surv}}$ finetuned on the data belonging to the target class. The target samples are used to generate an ‘inward pointing’( $\delta$ ) patch by solving an optimization problem to minimize the loss on the augmented data samples, which classifies as the target sample. . . . .	37
4.3	Visual distortion of an image as the magnitude ( $l_p$ norm) of augmentation increases. The leftmost image is the original image followed by augmentations with the rightmost image being the one with the highest augmentation ( $l_p$ ). . . . .	38
4.4	Visual Representation of various baseline augmentations to the target image . . . . .	40
4.5	The ROC curve . . . . .	41
4.6	The logarithmic scale ROC curve to find TPR at a low FPR . . . . .	42
C.1	Privacy - Utility curve for the loss based attacks . . . . .	65
C.2	Privacy - Utility curve for the modified entropy based attacks . . . . .	66
C.3	Privacy - Utility curve for the maximum confidence based attacks . . . . .	66
C.4	Privacy - Utility curve for the binary classifier based attacks . . . . .	67

# List of Tables

3.1	Ablation Study of $\pi_1$ only involved in L-Ctrl and $\pi_2$ only involved in C-Ctrl.	23
3.2	Sensitive analysis on the noise magnitude of target samples $\epsilon_2$ . Experiments are conducted on GTSRB with GMI attack. Injected samples use a magnitude of 8/255. Note that mislabel ratios are set to be $\pi_1 = 0$ , $\pi_2 = 0.5$ to amplify the effect brought by $\epsilon_2$ .	24
3.3	Overview of the attack methods, datasets, and models on which our method is evaluated.	25
3.4	Defense performance comparison against various MI attacks, where results are given in %. Note that for MIRROR, all 8 classes are target classes, and the classification accuracy is presented as ACC.	28
3.5	Defense performance against black-box MI - BREP-MI on two model architectures FaceNet64 and IR152	28
3.6	Defense performance against PPA on CelebA with different model architectures. Results are averaged over 6 randomly selected targets.	29
3.7	Defense performance against PPA on various datasets. Results are averaged over 6 randomly selected targets.	30
3.8	Defense performance with full mismatch and full match surrogate samples.	32
3.9	Impact of diversity and quality of surrogate samples within the same class.	33

4.1	Overview of the attack methods, datasets, and models on which the data-centric method is evaluated. . . . .	40
4.2	Overview of the augmentation methods for Metric Based Attacks - Confidence	43
4.3	Overview of the augmentation methods for Metric Based Attacks - Modified Entropy . . . . .	43
4.4	Overview of the augmentation methods for Metric Based Attacks - Loss . . .	44
4.5	Overview of the augmentation methods for Binary Classifier Attacks . . . .	44
4.6	Overview of the augmentation methods for Label-Only Attacks . . . . .	45
4.7	Overview of the augmentation methods for LiRA . . . . .	45
C.1	Privacy Parameters in DP-SGD and MID. . . . .	64



# List of Abbreviations

$f_\theta$  Classifier

L Loss

x Input

y Target class

AUC Area Under Curve

DNN Deep Neural Networks

DP-SGD Differential Privacy-Stochastic Gradient Descent

FPR False Positive Rate

GAN Generative Adversarial Networks

ML Machine Learning

ROC Receiver Operating Characteristic

TPR True Positive Rate

# Chapter 1

## Introduction

### 1.1 Motivation

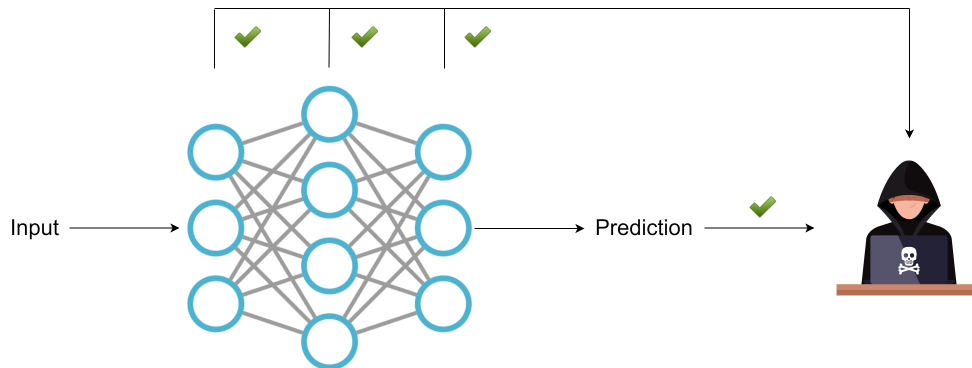
In recent years, the realm of Machine Learning (ML) has experienced a remarkable upswing across various domains, owing to a confluence of factors. This surge in popularity can be attributed to the exponential growth in computational power, the advent of open-source frameworks, and the wide availability of vast datasets. Within the vast landscape of ML, significant strides have been made in the specialized areas of natural language processing and computer vision, leading to unprecedented achievements.

Embracing this transformative technology, tech giants such as Microsoft, Google, and Amazon have taken the initiative to launch their own ‘Machine Learning as a Service’ (MLaaS) offerings, seamlessly integrating them into their cloud infrastructure. This innovative approach empowers users to effortlessly access and leverage the models, paying only for the specific services they utilize. However, amid these notable advancements, there remains a pressing need to shed light on the critical concerns surrounding ML privacy and security. Despite the immense potential and benefits offered by this technology, many individuals are still unaware of the associated challenges that demand immediate attention. Consequently, it is imperative to address these concerns promptly in order to ensure the responsible and secure implementation of ML systems.

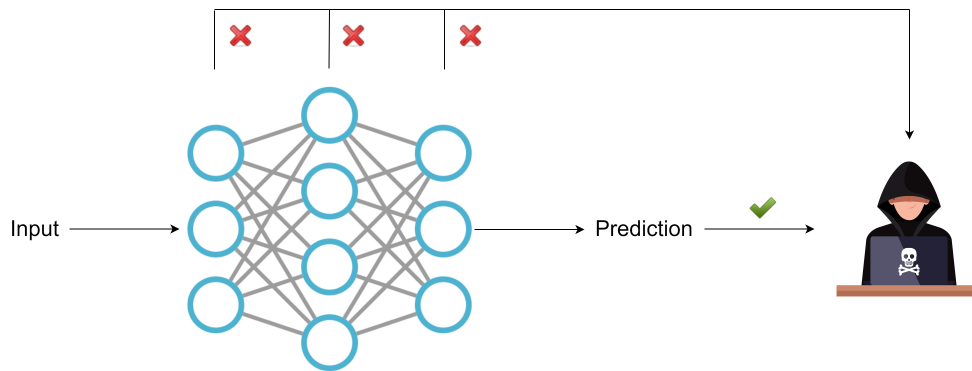
As a result of the increased usage of ML in our day-to-day activities, models have gained access to sensitive data like user images, videos, healthcare data, finances, etc. Regrettably, one aspect that has been frequently overlooked is the security of these models, leaving the stored data vulnerable to severe risks, including the inadvertent exposure of private information[6, 40, 52]. Contemporary studies have exhibited that ML models tend to memorize the data used in training, making this data available to adversaries via insecure ML models. Privacy Attacks typically make use of inference procedures to extract important information about the training data. Training a model is an iterative process whereby the model is fed with the training samples multiple times till it achieves the necessary performance. The intuition is that a ML model has distinguishable behavior on the data that it has seen iteratively (training data) versus the data it views for the first time (test data). Cloud platforms, due to their centralized nature and the storage of numerous ML models, are particularly vulnerable to security breaches. A single breach in the security infrastructure of a cloud service can potentially lead to widespread privacy violations, amplifying the urgency of addressing these vulnerabilities and implementing robust security measures.

**Adversary Knowledge** When considering privacy attacks on ML models, the level of knowledge or access that an adversary possesses about the target model is a crucial factor. Generally, there are two main categories to describe the extent of this knowledge represented in Figure.1.1: White box knowledge(Figure 1.1a) and Black box knowledge(Figure 1.1b).

**White-Box** In this setting, the attacker has unrestricted access to the information held by the target model. The adversary can know the model architecture, distribution of the training data, model parameters, and the training algorithm. White-box attacks assume that the attacker has full transparency into the model’s internal mechanisms and can use this knowledge to craft specific attacks.



(a) White-Box Knowledge: The adversary has the knowledge of the model along with the prediction



(b) Black-Box Knowledge: The adversary can only access the model prediction

Figure 1.1: Overview of White-Box attacks and Black-Box attacks

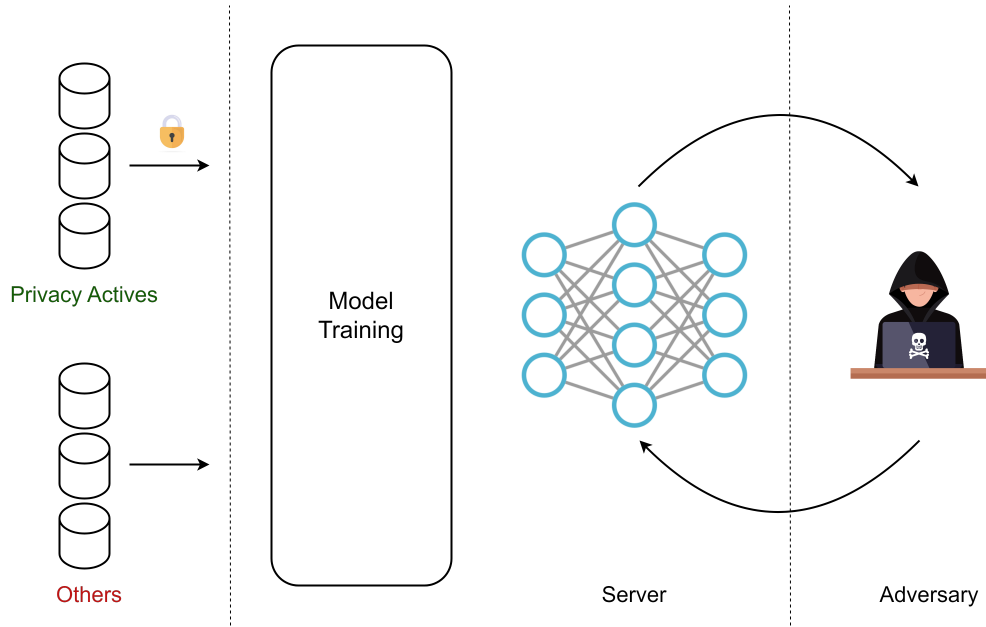
**Black-Box** The adversary only has black-box access to the model. The adversary has limited information and has query access to the model e.g. the adversary queries the model to get the model predictions. Black-box attacks aim to infer sensitive information or exploit vulnerabilities by observing the model's behavior and responses without having direct access to its internal details.

Existing defenses against privacy attacks mainly apply a model-centric approach, wherein the model owner tries to revise the training algorithms or inference procedures such that data privacy is protected. Privacy Guarantee techniques like DP-SGD (Differentially Private Stochastic Gradient Descent)[1] is one of the commonly used methods.

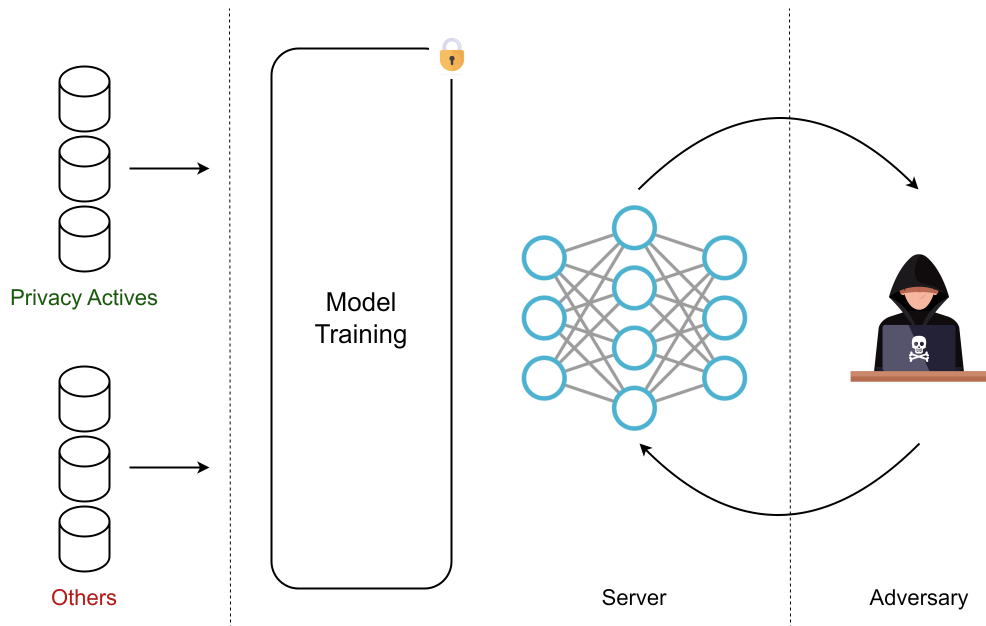
In DP-SGD, the model owner modifies the training by clipping followed by adding noise to the training gradients. MemGuard [20] is a popular membership inference defense that changes the inference process by adding noise to the model prediction vector. However, it is important to acknowledge that these contemporary techniques place the burden of data protection solely on the model owner. In essence, data owners are compelled to place complete trust in the model owner’s ability to safeguard their sensitive information.

Consequently, this heavy reliance on the model trainer for ensuring privacy significantly restricts the user’s control over their own data. Users become highly dependent on the model trainer’s expertise and actions to protect the confidentiality of their sensitive information. Moreover, the approach of altering the model training process to enhance privacy has notable implications on the model’s overall performance, often resulting in reduced accuracy or increased computational time. Recognizing these limitations and challenges imposed by model-centric defenses, our work is motivated by the need to address these bottlenecks. The aim of this work is to explore alternative approaches that provide users with greater control over their data privacy while minimizing the impact on model performance.

Data collection is one of the most important aspects of model training. One of the common ways to collect data is to use web crawlers or approach individual data owners to gather data and create a dataset. The concerns about data privacy have resulted in legislations like the California Consumer Privacy Act[34] and GDPR[29] which advocate for individuals to have complete authority over their data including the prerogative to withdraw their data from existing datasets. In alignment with these legislative efforts and the growing need for enhanced data privacy, our approach focuses on empowering data owners in a decentralized manner. We aim to develop methodologies that enable individuals to have greater control and ownership over their data, ensuring that their privacy is protected throughout the data lifecycle.



(a) Data Centric Defense: Data is augmented by the data owner using privacy-focused data augmentations before sending it to the model owner for training



(b) Model Centric Defense: The model trainer is responsible to provide data protection by changing the model

Figure 1.2: Data Centric Defense vs Model Centric Defense

## 1.2 Thesis Contributions

In this thesis, our primary interest is in analyzing and addressing the prevention of attacks on data privacy, specifically membership inference attacks and model inversion attacks, through the implementation of our data-centric defense strategy, as shown in 1.2. The contributions can be summarized as follows:

- Analyzed Model Inversion and Membership Inference Attacks
- Studied and analyzed Model Inversion and Membership Inference Defenses to formulate data-centric defense strategies
- **Privacy Focused Data Augmentations** Formulated and developed data augmentations focused on providing data privacy along with an improvement in the model generalizability
- **Data-Centric Defense** Implemented a data-centric defense against privacy attacks with no dependence on the model trainer to provide data protection, giving complete authority to the data owner to protect one's own data
  1. Data-Centric Defense against Model Inversion Attacks
  2. Data-Centric Defense against Membership Inference Attacks

The thesis attempts to enhance the security and privacy of ML systems, enabling users to have greater confidence in the protection of their sensitive data.

## 1.3 Thesis Outline

The thesis organization is as follows: Chapter 2, contains the survey of the literature related to privacy attacks and existing techniques to mitigate them. Chapter 3 is a detailed study of the data-centric defense for model inversion[9] containing the methodology and the outcome of the data-centric defense. Chapter 4 contains an extension of the work to defend against membership inference attacks, using a data-centric approach. The final chapter, Chapter 5 has the conclusion and future work.



# Chapter 2

## Review of Literature

### 2.1 Machine Learning

Machine Learning (ML) refers to the realm of computer algorithms that harness the power of data to enable machines to acquire knowledge through experience[30]. Model training is a crucial aspect of ML. Given a model denoted as  $f_\theta$ , the training process involves learning the parameters associated with  $f_\theta$  in order to accurately map the input  $x$  to the corresponding output  $y$ . The primary objective of training an ML model is to minimize the prediction loss on the inputs.

### 2.2 Privacy Attacks

Privacy attacks specifically target the extraction of sensitive information from candidate samples by exploiting security vulnerabilities within the trained models [35]. This section is directed toward the examination of privacy attacks, with a particular emphasis on membership inference and model inversion attacks. Understanding these attacks allows for the development of more robust defense mechanisms and privacy-preserving techniques to safeguard sensitive data and protect individuals' privacy.

## 2.3 Model Inversion

The model inversion attack, as introduced by Fredrikson[14], involves inverting an ML model. In this method, the attacker aims to reconstruct a training sample that accurately represents a specific class, as depicted in Figures 2.1a and 2.1b.

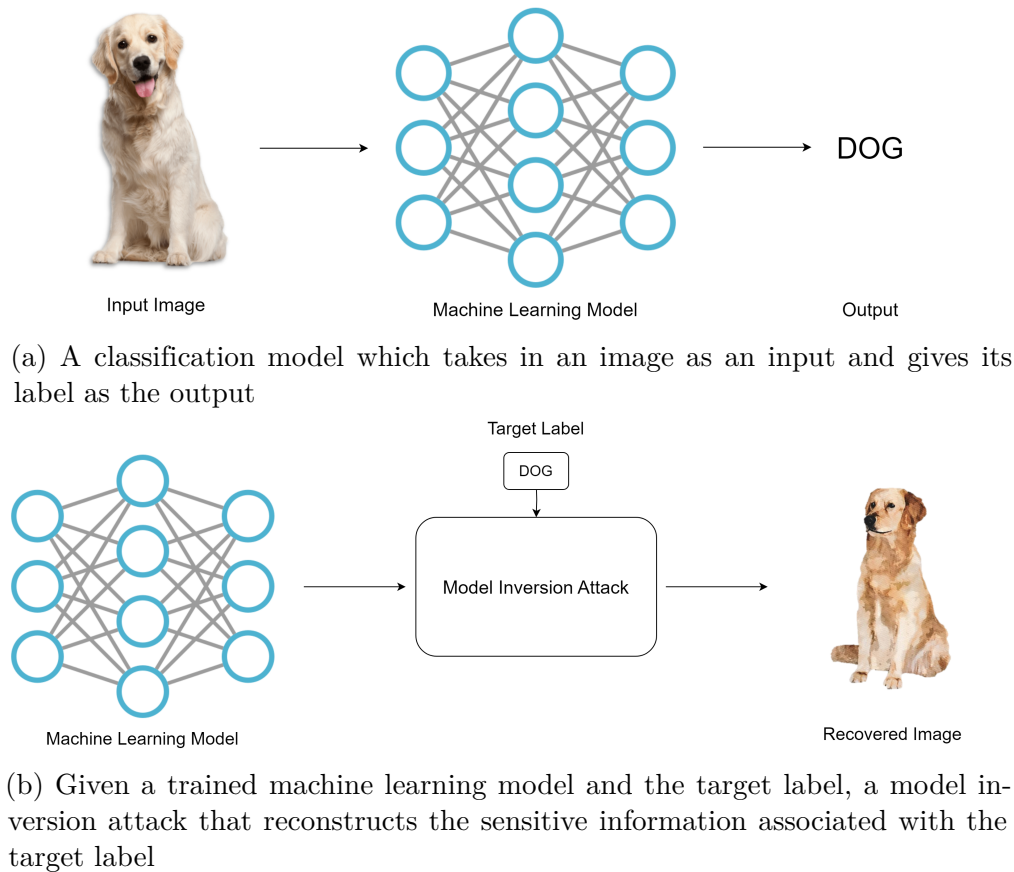
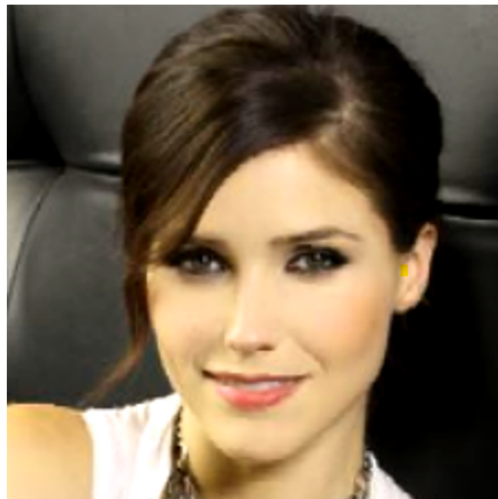


Figure 2.1: Illustration of a model training algorithm and a model inversion attacks

In recent research, there has been a specific emphasis on face-recognition tasks that exploit models to identify facial attributes associated with a given individual (output label). This work also concentrates on face recognition, where the adversary’s objective is to reconstruct facial images pertaining to a specific class. Model inversion attacks[2, 45, 54] solve an optimization problem to minimize loss or maximize the likelihood for the target model.



(a) Recovered image



(b) Original image

Figure 2.2: An example of the image recovered using model inversion attack. Compared with a sample image belonging to the target class, the recovered image showcases similarity in the facial features

### 2.3.1 Model Inversion Attacks

The initial model inversion attack, as described by [14], aimed to recover genomic privacy. This attack employed a linear regression model, auxiliary information, and the prediction vector obtained through model inference. By solving an optimization problem, these attacks identified the sensitive features. However, these techniques were limited to simple linear models. In the case of deep neural networks (DNNs), the optimization process can become non-convex, resulting in inconsistent outcomes.

An alternative approach called GMI[54] utilizes a generative attack based on Generative Adversarial Networks (GAN)[16] with white-box access to the target DNN model. The authors leverage publicly available datasets that share the same distribution as the target dataset to generate images that serve as priors for model inversion attacks. This attack optimizes the latent space to produce samples resembling the desired target class. However, GMI is only effective for inverting low-resolution images (64x64).

In contrast to traditional GAN techniques, MIRROR[2] employs a specialized architecture based on StyleGAN[22], pre-trained on a publicly available dataset that is shared across domains. This architecture separates inputs into different styles at various levels, enabling more effective model inversion. By utilizing only one pre-trained StyleGAN2[23], PPA[45] achieves disentanglement in the latent space which is optimized for implementing model inversion Attacks.

Unlike white-box attacks, where the attacker has access to the entire model architecture, the authors of BREP-MI[21] propose an attack in a more practical scenario. In this scenario, the adversary only has access to the predicted label rather than the complete confidence vector.

### 2.3.2 Model Inversion Defenses

Model-specific defenses primarily concentrate on adjustments made to the training algorithm. One effective technique for privacy-preserving ML and defense against model inversion attacks is DP-SGD (Differentially Private - Stochastic Gradient Decent)[1]. Empirical evidence demonstrates that DP-SGD can assist in mitigating attacks by injecting a sufficiently large amount of noise. However, it should be noted that injecting excessive noise can lead to a negative impact, causing a deterioration in the performance of the model.

[47] conducted a study on the theoretical underpinnings of the limited effectiveness of DP-SGD in mitigating model inversion attacks. They also proposed an alternative approach by incorporating information bottleneck-based learning objectives to reduce the correlation between model outputs and training data. This involved modifying the model architecture and adjusting the training process. Although this method yielded improved results compared to DP-SGD, it still faced challenges related to the privacy-utility tradeoff. Additionally, it introduced an overhead by necessitating changes to the model architecture.

In contrast to model-centric defenses, which do not provide user-level privacy, the proposed technique solely relied on data augmentation carried out by privacy-conscious individuals, referred to as “privacy active”.

## 2.4 Membership Inference

The concept of membership inference was initially introduced by Homer[18] to address the identification of an individual’s genome within a collection of genomes. This was achieved by comparing data statistics. Subsequently, the idea of membership inference attacks in the ML domain was proposed by [38]. In their work, the authors aimed to determine which specific data samples belonged to the training set.

ML models are trained on data for multiple instances or epochs, which can lead to the memorization of those specific samples[15]. Furthermore, training data may not fully represent the entire data distribution, resulting in poor generalization of the models. Consequently, ML models may exhibit distinct behavior when applied to training data versus unseen test data. Overfitting is considered a key factor contributing to the success of membership inference Attacks[38, 49].

Figure 2.3 shows how a membership inference attack works. Given a set of data points, membership inference attacks aim to identify samples from the training set. The data points are queried to the target model to get the model output. This output is then used by the membership inference attack algorithm to

Researchers continue to explore and develop new strategies to improve the effectiveness and understanding of these attacks.

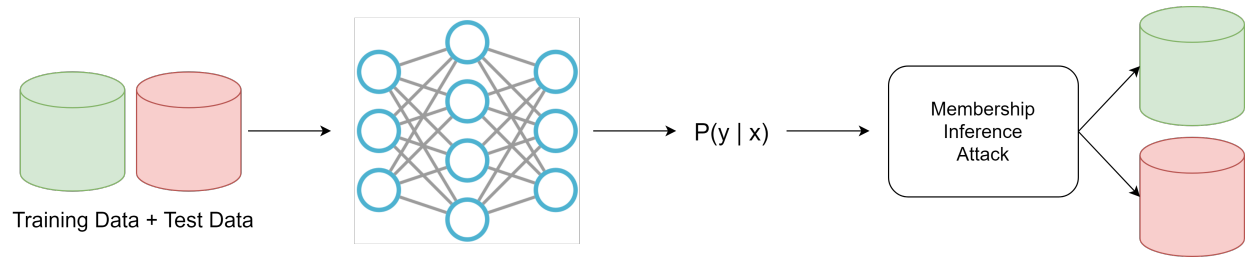


Figure 2.3: Overview of a Membership Inference Attack. A set of data samples are passed through a trained ML model. The output is then given as an input to the membership inference attack which distinguishes between the training samples and the non-training samples.

### 2.4.1 Membership Inference Attacks

Membership inference attacks are indeed significant security threats, particularly when it comes to sensitive data. For instance, these attacks can reveal the identity of a healthcare record that has been utilized in a training dataset. By leveraging membership inference attacks, it is possible to infer the owner of such data with a high level of accuracy. The attacks can be used for data extraction, model impersonification, and auditing. These attacks can generally be categorized into four types:

- Binary Classifier Based
- Metric Based
- Likelihood Ratio Based
- Label-Only Based

## Binary Classifier Based Attacks

[38] used a binary classifier for membership inference where several shadow models were used to imitate the behavior of the target model. The shadow model outputs are used to train a binary attack classifier, to predict the membership of data points on a target model. [37] proposed to relax the assumptions of [38] by reducing the number of shadow models to one. [49] suggests membership on the basis of a correct prediction by the target model. A misclassified sample can be attributed to a test sample as the model will have 100% classification accuracy on the data it has been trained on. Thus, the output of the classifier is used as a proxy to identify the membership status of a sample.

## Metric Based Attacks

These types of attacks do not rely on a binary classifier to determine the membership status. Rather, they require less computation and make use of thresholding based on average case/sample-specific metrics. Some of the metrics are as follows:

- 1. Correctness** An attacker infers a member if the data record is correctly predicted by the target model [49]. The logic is that since the model is well-trained on the training data, it fails to generalize on unseen data and thus misclassifies it.
- 2. Loss** [36] provides further strength to [49]’s theory to propose that model loss is the only factor and provides the approximations for optimal strategy. Classification models are trained to resolve an optimization problem to minimize the loss. Thus, members showcase a lower loss than non-members.

**3. Entropy** [37] concludes that since a model is trained to minimize the loss over training data results in the prediction output of training samples close to its one-hot encoded label vector. Thus, the prediction entropy of members is almost equal to 0, way lower than non-members. [42] proposed a modified version of entropy considering the ground truth.

**4. Confidence** Since ML models are trained over multiple epochs on training data, models tend to have higher confidence in member samples given by a higher value in the prediction vector corresponding to its label.

### Likelihood Ratio Based Attacks

[7] first trained  $N$  shadow models  $P = \{\theta_1, \dots, \theta_N\}$  splitting the dataset  $D$  randomly such that a sample point  $(x, y) \in D$  belongs to the training set of  $N/2$  models (denoted as *IN* models) and outside the training set of  $N/2$  models (denoted as *OUT* models). For a sample point  $(x', y')$ , the adversary measures the respective confidence scores for *IN* and *OUT* models given by  $P_{in} = \{\theta_1^{in}, \dots, \theta_{N/2}^{in}\}$  and  $P_{out} = \{\theta_1^{out}, \dots, \theta_{N/2}^{out}\}$

$$\phi(f_{\theta}(x)_t) = \log\left(\frac{f_{\theta}(x)_t}{1 - f_{\theta}(x)_t}\right) \quad (2.1)$$

where  $f_{\theta}(x)_t$  is the confidence vector of a sample point  $x$  in target class  $t$ . This is used to generate logit scores for *IN* and *OUT* models and fits two Gaussian distributions  $\mathcal{N}(\mu_{in}, \sigma_{in}^2)$  (*IN*) and  $\mathcal{N}(\mu_{out}, \sigma_{out}^2)$  (*OUT*). When the attacker queries for a point  $(x, t)$ , the logit score is found to estimate the membership probability.

$$\text{membership} = \frac{p(f_{\theta}(x)_t) | \mathcal{N}(\mu_{in}, \sigma_{in}^2)}{p(f_{\theta}(x)_t) | \mathcal{N}(\mu_{out}, \sigma_{out}^2)} \quad (2.2)$$



## Label-Only Attacks

Contemporary attacks assume the adversary to have the knowledge of the entire prediction vector, however in most cases, the model only returns the prediction label for the query. [12, 26] came up with attacks based on the prediction label. Both attacks use a fundamental strategy of evaluating membership based on the L2 distance of the sample and the decision boundary. A higher magnitude perturbation is necessary to cause mislabeling for members compared to non-members.

### 2.4.2 Membership Inference Defenses

Numerous defense mechanisms have been proposed to combat membership inference attacks. Some of these defenses focus on mitigating overfitting in ML models and safeguarding the privacy of the data.

By addressing overfitting, these defenses aim to prevent attackers from extracting sensitive information about the membership status of specific data samples. Overfitting reduction techniques such as regularization, early stopping, and dropout can help enhance the model's generalization capabilities and make it less susceptible to membership inference attacks.

Additionally, other defenses involving privacy-preserving changes to training/inference pipelines are also described below.

#### 1. Regularization

It is a technique to reduce model complexity and reduce overfitting. Using the L2 norm which penalizes the larger parameters by adding an additional term  $\lambda \sum \theta_2$ , a larger  $\lambda$  has a stronger effect during training.

## 2. Dropout

Overfitting causes the model memorizes the training data so well that it starts to learn the noise as well making the model have a poor generalization performance. In the context of Dropout, this technique addresses overfitting by selectively dropping or ignoring certain nodes in the network layers during training. By doing so, Dropout disrupts the connectivity within the network, preventing specific nodes from relying too heavily on others. This regularization technique helps to reduce the model's reliance on individual nodes and encourages the network to learn more robust and generalized representations, thus improving its ability to generalize to unseen data.

## 3. Data Augmentation

Data augmentation is a technique employed to enhance the performance of a model on unseen data. It involves applying various transformations to the existing training data and creating additional synthetic examples. By diversifying the training data through augmentation, the model becomes more adept at handling variations and generalizing better to unseen instances, leading to improved overall performance. However, only certain data augmentation techniques help reduce the train-test gap making the data resilient to attacks [24].

## 4. Early Stopping

Early stopping is a technique utilized to mitigate overfitting without sacrificing the accuracy of the model. It involves terminating the training process before the model starts to exhibit signs of overfitting, thus improving its generalization capabilities and overall performance. [41] recommends early stopping to perform better than the proposed defenses [20, 32].

## 5. Differential Privacy - Stochastic Gradient Descent(DP-SGD)

DP-SGD[1] provides privacy guarantees for the data to protect it from MI Attacks. DP-SGD aims at clipping the gradient and adding noise to them thereby reducing the influence of individual training samples on the data. However, doing this affects the model performance to a great extent.

## 6. RelaxLoss

Membership inference attacks often rely on distinguishing between the training and test loss to determine membership status. To address this, RelaxLoss[8] is introduced to minimize the gap between these losses and mitigate model leakage. RelaxLoss works by altering the loss distribution through gradient ascent, transforming the optimization problem to include a non-zero training loss. This defense mechanism aims to balance the preservation of utility by implementing techniques such as posterior flattening and gradient normalization.

## 7. Adversarial Regularization

A surrogate model is employed to approximate the MI attack, and this information is utilized as a regularization term during the training of the target model[32]. The process involves a two-step optimization approach: the inner maximization occurs on the surrogate model to perform membership inference, while the outer minimization aims to identify the most robust classification model for addressing the inner optimization problem.

## 8. MemGuard

In MemGuard[20], the prediction vector of the target classifier is manipulated by introducing noise in a manner that maintains the same predicted label for the data sample. This strategy serves the purpose of deceiving the attacker by transforming the query into an adversarial sample that is designed to mislead them.

## 2.5 Relation between Data Augmentation and Privacy

Data augmentation is a widely employed technique in ML, which entails applying various transformations or modifications to existing data samples, thereby artificially expanding the training dataset. The objective of data augmentation is to introduce greater diversity and variability into the training data, leading to improved model generalization and overall performance in ML tasks. While the benefits of data augmentation for enhancing model performance have been extensively studied, its implications for privacy and vulnerability to membership inference attacks[10, 24, 46] have also been investigated.

The effects of data augmentations on model inversion attacks, compared to membership inference attacks, have received less attention in research. [24] investigated whether augmentations, which are typically employed to enhance model generalization, also provide privacy benefits against membership inference attacks.

This thesis goes beyond the traditional belief of augmentations improving model performance to providing a privacy guarantee to data and protect data from both membership inference and model inversion attacks.

# Chapter 3

## Model Inversion

Model inversion attacks aim to reconstruct the sensitive features belonging to the samples of a particular target class. This is done by solving an optimization problem that reconstructs the samples with the highest likelihood or the lowest loss for the target model. The proposed method aims to preserve target samples and influence the Model Inversion Attack to recover the surrogate samples. The method provides the data owner with the tools to ensure protection for their data without relying on the model trainer to provide protection.

### 3.1 Proposed Methodology

Let  $f_\theta$  be the target model mapping inputs  $x \in X$  to label  $y \in Y$  where  $Y = \{y_1, \dots, y_m\}$ . Unprotected training set is denoted by  $D = \{(x_{ij}, y_i)\} : i = 1, \dots, m, j = 1, \dots, k$  where  $x_{ij}$  represents the  $j^{th}$  samples in class  $i$  where  $k_i$  is the number of samples in class  $i$ . For a face recognition problem,  $y_i$  represents a distinct person, and  $x_{ij}$  is a sample belonging to it. The method aims to protect these samples indexed by  $S_{tgt}$  called the target label set. The detailed pseudocode is given in [Appendix B.1](#).

### Injection of surrogate samples to the training set

The first step is to identify a surrogate class such that it does not disclose any sensitivity of the target class. The first step is to collect all samples from the surrogate class ( $x_j^1, j = 1, \dots, m; x_j^1 \sim P(X|y_{srg})$ ) and relabel them as the target class. The result is an augmented dataset given by

$$D_{y_{tgt}}^1 = \{(x_j^1, y_{tgt}) : j = 1, \dots, m\} \quad (3.1)$$

The model is thus trained on a combination of augmented target and surrogate samples, both labeled as the target class. This creates confusion for the adversary about the correctness of the attributes belonging to the target class. However, only injecting surrogate samples is not enough as our aim is to prevent the reconstruction of the target samples.

### Loss-Controlled Modification for target samples

Model inversion attacks solve an optimization problem arriving at samples with the lowest loss for the prediction loss. To protect the target samples, the loss of target samples must be higher than the surrogate samples. This results in the recovery of the surrogate samples (lower loss) over target samples when queried for a target class. Following this, a small fraction ( $\pi_1$ ) of the target samples is randomly mislabeled, increasing the loss associated with target samples, and keeping the surrogate samples having the label as target class.

$$D_{y_{tgt}}^0 = (x_j^0, y_j') : j = 1, \dots, \lceil m\pi_1 \rceil \cup (x_j^0, y_{tgt}) : j = \lceil m\pi_1 \rceil + 1, \dots, m \quad (3.2)$$

where  $x_j^0 \sim P(X|y_{tgt})$  and  $y_j' \sim \text{Uniform}(Y \setminus y_{tgt})$

### Injection of samples to shape the Loss Curvature

The loss modification aids the surrogate injection in improving the defense performance. However, a side effect of this is a reduced model performance (up to 5-7%). Relying on non-convex optimization theory[4], the loss landscape’s curvature is influenced by stimulating a flatter curvature around the surrogate samples while a steeper curve around target samples. This is achieved by adding Gaussian augmentation to the surrogate samples while keeping the same label.

$$D_{y_{tgt}}^2 = (x_j^1 + \mu_j, y_{tgt}) : j = 1, \dots, m \quad (3.3)$$

where  $\mu_j \sim \mathcal{N}(0, \epsilon_1^2)$ .

For target samples, however, a portion ( $\pi_2$ ) of samples is mislabelled after adding Gaussian augmentation.

$$D_{y_{tgt}}^3 = (x_j^0 + \mu'_j, y'_j) : j = 1, \dots, \lceil m\pi_2 \rceil \cup (x_j^0 + \mu'_j, y_{tgt}) : j = \lceil m\pi_2 \rceil + 1, \dots, m \quad (3.4)$$

where  $\mu'_j \sim \mathcal{N}(0, \epsilon_2^2)$  and  $y'_j \sim \text{Uniform}(Y \setminus y_{tgt})$

The model memorizes the training samples which yield different labels for the target samples and their augmented neighbors. This distorts the reconstruction toward generating surrogate samples. Figure 3.1 is representative of the methodology to shape the loss curve.

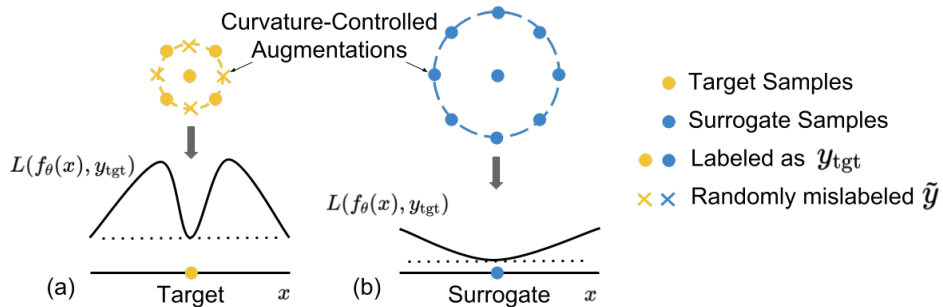


Figure 3.1: Illustration of curvature-controlled augmentations and the resulting loss landscape of target and surrogate samples.

### Analysis on Mislabeling Ratio ( $\pi_1$ ) and ( $\pi_2$ )

Solely injecting surrogate samples in the training set does not effectively mitigate the risk of MI attacks. However, when combined with either loss control or curvature control, the attack accuracy decreases to approximately 10%. By employing all three techniques together, the attack accuracy is reduced to 0.0%. An ablation study on parameter selection is presented in Table 3.1.  $\pi_1$  is the mislabel ratio used for the Loss-Controlled modification whereby a portion ( $\pi_1$ ) of the target samples are mislabeled and  $\pi_2$  is the ratio of mislabeled samples after adding Gaussian augmentation for controlling the loss curvature.

	No Protection	Surr-Inj	Surr-Inj&L-Ctrl			Surr-Inj&C-Ctrl				Surr-Inj&L-Ctrl&C-Ctrl		
Mislabel Ratio ( $\pi_1$ )	-	-	0.1	0.2	0.5	-	-	-	-	0.1	0.2	0.2
Mislabel Ratio ( $\pi_2$ )	-	-	-	-	-	0.1	0.2	0.5	1.0	0.5	0.5	1.0
ACC-all	98.58	98.46	98.14	97.98	97.89	98.50	98.62	97.87	97.86	98.39	97.97	97.96
ACC-tar	99.25	100.00	98.45	97.97	95.15	99.42	99.71	98.55	98.51	98.99	97.94	97.38
Att. ACC	79.20	29.60	12.60	9.80	0.60	21.80	19.80	11.80	10.60	0.30	0.00	0.00

Table 3.1: Ablation Study of  $\pi_1$  only involved in L-Ctrl and  $\pi_2$  only involved in C-Ctrl.

### Analysis on Noise Magnitude of Target Samples $\epsilon_2$

This is a supplementary experiment to investigate the influence of different noise magnitudes on target samples  $\epsilon_2$ . It is important to note that, throughout this thesis, a fixed noise magnitude ( $\epsilon_1 = 8/255$ ) is maintained for all experiments. Selecting  $\epsilon_2$  values that are smaller than  $\epsilon_1$ , can further enhance the control strength and create sharper curvature in the target samples. As expected, the results in Table 3.2 demonstrate that our method achieves comparable and satisfactory performance when using  $\epsilon_2 < 8/255$ , with the best performance observed at  $\epsilon_2 = 0.003$ . On the other hand, for  $\epsilon_2 > 8/255$ , the strength of curvature control weakens, resulting in a lower defense performance, where the attack accuracy is around 30%.



	Gaussian Noise Magnitude $\epsilon_2$						
	0.001	0.003	0.005	0.01	8/255	0.1	0.3
<b>ACC-all</b>	97.75	97.21	98.16	97.458	98.12	97.32	97.32
<b>ACC-tar</b>	99.13	98.99	95.57	99.71	99.13	99.86	99.57
<b>Att. ACC</b>	2.60	0.40	2.00	2.20	5.80	26.20	35.40

Table 3.2: Sensitive analysis on the noise magnitude of target samples  $\epsilon_2$ . Experiments are conducted on GTSRB with GMI attack. Injected samples use a magnitude of 8/255. Note that mislabel ratios are set to be  $\pi_1 = 0$ ,  $\pi_2 = 0.5$  to amplify the effect brought by  $\epsilon_2$ .

## 3.2 Experiments

This section contains a discussion of the experimental setup for the algorithm to answer questions related to the generalizability of the technique over different models and datasets, comparison with existing defenses, and the choice of hyperparameters and surrogate samples.

### 3.2.1 Attack Algorithm

The effectiveness of the proposed data-centric defense is evaluated on three white-box model inversion attacks: GMI, PPA, and MIRROR-W. PPA and MIRROR are the more recent attacks built over the initial idea of using GANs by GMI. To carry out a thorough evaluation of the method, it is also evaluated over black-box techniques like the black-box counterpart of MIRROR-W called MIRROR-B and BREP-MI.

### 3.2.2 Evaluation Setup - Attacks, Models, and Datasets

The efficiency of data-centric defense across different models and datasets utilized in various model inversion attacks is studied in this section. Table 3.3 gives the details of the different datasets and models used for evaluation.

Attack Method	Task	Private Dataset	Public Dataset	Pre-trained GAN	Model
GMI	Traffic Sign Recognition	GTSRB[44]	TSRD	WGAN [3]	VGG-16[39]
PPA	Face Recognition	CelebA[27]	FFHQ MetFaces	StyleGAN2 <sup>1</sup> [23]	ResNet-152[17], ResNext-101[48], DenseNet-169[19], ResNeSt-101[53]
		FaceScrub[33]	FFHQ MetFaces	StyleGAN2	ResNeSt-101 ResNeSt-101
	Dog Classification	St.Dogs[25]	AFHQ	StyleGAN2	ResNeSt-101
MIRROR-B	Face Recognition	CelebA-partial256	VGGFace2[5]	StyleGAN <sup>2</sup>	VGG-16
MIRROR-W	Face Recognition	CelebA-partial256	VGGFace2	StyleGAN[22]	VGG-16
BREP-MI	Face Recognition	CelebA	CelebA	WGAN	face.evoLVe[11], IR-152[17]

Table 3.3: Overview of the attack methods, datasets, and models on which our method is evaluated.

The public and private datasets are decided based on their tasks consisting: 1. Face Recognition (Private: CelebA, FaceScrub, Public: FFHQ, MetFaces); 2. Traffic Sign Recognition (Private: GTSRB, Public: TSRD); 3. Dog Classification (Private: St. Dogs, Public: AFHQ) for model inversion attacks GMI, PPA, MIRROR, and BREP-MI. The attacks employ generative methods to devise an image prior, using different GANs for different attacks pre-trained on public datasets sharing the same distribution as the private dataset. The detailed analysis of datasets and models is presented in Appendix A.

### 3.2.3 Comparison with Baseline Defenses

A comparison with the existing defenses of DP-SGD and MID is made in this section. To ensure the reproducibility and consistency of results, the open-sourced implementations of both these techniques [47] have been used. The privacy parameters (hyperparameters) associated with ensuring adequate privacy are configured by running multiple variations of each baseline method. The choice of hyperparameters or ‘privacy parameters’ is shown in Appendix C.1

**DP-SGD** DP-SGD clips and adds noise to the training gradient. The privacy parameters include the gradient clipping threshold  $C$ , noise multiplier  $\sigma$  adjusted to get the required privacy budget  $\epsilon$ , and probability upper bound  $\delta$ . The  $\delta$  has been set to a value equivalent to  $\frac{1}{\text{size of data}}$ . The learning rate and the batch size are the same as the unprotected version.

**MID** MID restricts the information from the model prediction. MID adds a regularization term to the loss known as information loss to achieve privacy. MID introduces  $\beta$  (the weight given to the information loss) to reduce the correlation between the prediction and the input.

### 3.3 Results

Figure 3.2 displays a comparison between the recovered images for the baseline defenses by the Model Inversion Attack. Figure 3.3 displays the surrogate sample to be injected for the corresponding target samples (Figure 3.3a) which successfully deceives the attacks to generate representative samples of the target class resembling the injected surrogate samples. (Figure 3.3b)

**Interpretation of Results** The following metrics are to be considered in order to understand the results.

- **ACC-all** Classification accuracy over the entire test set, represented in %. The ACC-all should be high for the ML model to perform well on the downstream task at hand.
- **ACC-tar** Classification accuracy over the target test samples, represented in %. The ACC-tar should be high for the ML model to perform well on the test-target samples.

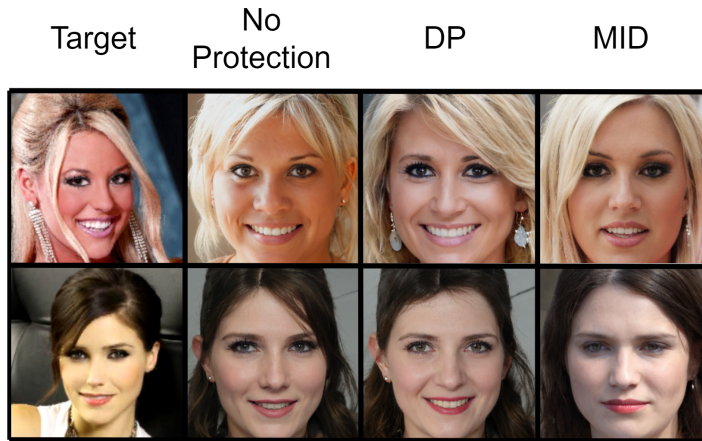


Figure 3.2: Visual representation comparing the faces recovered for different baselines by the Model Inversion Attack. Each row shows the reconstruction of an image belonging to the same identity. The true image is on the left followed by the reconstruction under no protection, DP-SGD, and MID

- **Att.ACC** Attack Accuracy determines the success of the Model Inversion Attack. Given in %, the value should be 0.00% to achieve perfect protection of data samples.

### Performance against various attacks

A detailed comparison of our method is done with existing defenses on different attacks, varying the model architectures and datasets. The results are averaged over multiple classes repeating over multiple iterations to ensure consistency leaving out the stochasticity introduced by the model training. Table 3.4 reflects the comparison of the performance with various levels of protection. The unprotected model showcases a high attack accuracy (Att. ACC) of 100% for MIRROR-W as well as MIRROR-B followed by 90% for PPA, 76% for GMI. While the attack accuracy is reduced, both MID and DP-SGD show a significant loss in classification accuracy. On the other hand, the data-centric method significantly reduces the attack accuracy to 0% for MIRROR-W, MIRROR-B, and GMI, and 1% for PPA. Thus, a better privacy-utility balance is ensured by maintaining utility while upholding privacy.

	GMI TSRD→GTSRB			MIRROR-W FFHQ→VGGFace2	
	ACC-all	ACC-tar	Att. ACC	ACC	Att. ACC
<b>No Protection</b>	98.34	99.20	76.13	99.99	100.0
<b>DP</b>	54.30	31.24	12.80	56.25	54.69
<b>MID</b>	67.70	55.37	54.53	41.34	100.00
<b>Data-Centric Defense</b>	95.89	93.74	0.00	96.88	0.00

	PPA FFHQ→CelebA			MIRROR-B FFHQ→VGGFace2	
	ACC-all	ACC-tar	Att. ACC	ACC	Att. ACC
<b>No Protection</b>	88.42	84.37	90.40	99.99	100.0
<b>DP</b>	39.61	6.67	14.33	56.25	50.00
<b>MID</b>	69.54	53.33	52.33	41.34	12.50
<b>Data-Centric Defense</b>	88.05	81.88	1.00	96.88	0.00

Table 3.4: Defense performance comparison against various MI attacks, where results are given in %. Note that for MIRROR, all 8 classes are target classes, and the classification accuracy is presented as ACC.

The performance is computed also on the label-only black-box attack BREP-MI. The assessment is carried out using the same implementation settings as BREP-MI with two datasets FaceNet64 and IR152. Consistent with the earlier evaluation, 6 target classes are selected for evaluation. The results can be seen below in Table 3.5

	FaceNet64			IR152		
	ACC-all	ACC-tar	Att.ACC	ACC-all	ACC-tar	Att.ACC
<b>No Protection</b>	86.78	93.33	83.33	89.05	81.87	66.67
<b>Data Centric Defense</b>	85.72	85.86	0.00	92.31	86.67	0.00

Table 3.5: Defense performance against black-box MI - BREP-MI on two model architectures FaceNet64 and IR152

### Generalization over different Models

In this part, the performance on different DNN models is evaluated. The selected models are ResNet-152, ResNext-101, ResNeSt-101, and DenseNet-169. The results in Table 3.6 draw attention to the model-agnostic ability of our method. The data-centric defense successfully negates the attack with the attack accuracy reaching levels less than 5% in spite of a high attack potency of PPA on unprotected samples. The data-centric approach thus preserves privacy irrespective of the model architecture and training method.

	ACC-all	ACC-tar	Att. ACC	ACC-all	ACC-tar	Att. ACC
	<b>ResNeSt-101</b>			<b>ResNet-152</b>		
<b>No Protection</b>	88.42	84.37	90.40	84.82	80.00	76.67
<b>Data Centric Defense</b>	88.05	81.88	1.00	85.33	86.67	4.00
	<b>DenseNet-169</b>			<b>ResNext-101</b>		
<b>No Protection</b>	84.85	60.00	73.67	85.89	73.33	84.67
<b>Data Centric Defense</b>	84.32	60.00	3.00	87.16	60.00	2.00

Table 3.6: Defense performance against PPA on CelebA with different model architectures. Results are averaged over 6 randomly selected targets.

### Generalization over different Datasets

The performance was assessed on the PPA attack (Table 3.7) with diverse datasets representing different tasks - CelebA, FaceScrub (both face recognition), and St.Dogs (dog classification). For a thorough evaluation, GANs were pre-trained on FFHQ and MetFaces for facial datasets and AFHQ as the public dataset for St.Dogs. FFHQ-based GAN was more effective with an attack accuracy of 90% against 59% for MetFaces for the unprotected sample set. Data-centric defense works effectively with the attack accuracy going down to 1% and 0.02% respectively.

CelebA				
	ACC-all	ACC-tar	FFHQ	MetFaces
			Att.ACC	Att.ACC
<b>No Protection</b>	88.42	84.37	90.40	59.33
<b>Data Centric Defense</b>	88.05	81.88	1.00	0.02

FaceScrub				
	ACC-all	ACC-tar	FFHQ	MetFaces
			Att.ACC	Att.ACC
<b>No Protection</b>	95.78	97.50	82.40	53.20
<b>Data Centric Defense</b>	94.93	90.37	1.20	4.20

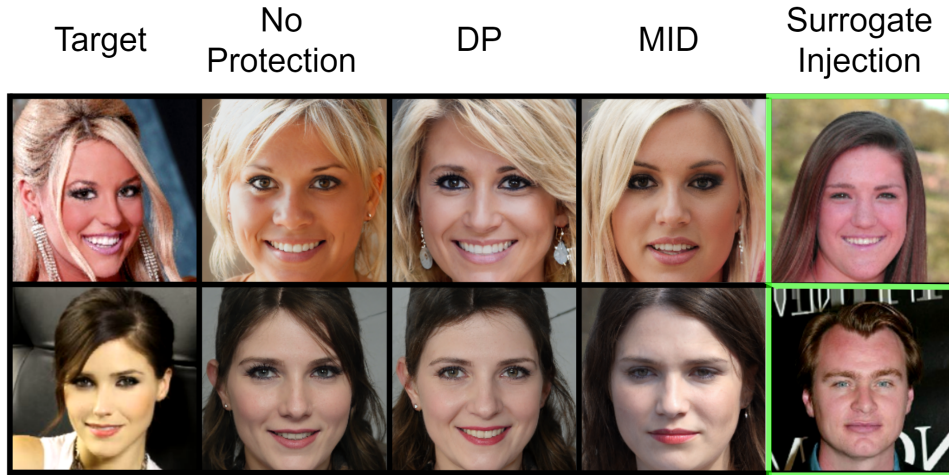
St.Dogs			
	ACC-all	ACC-tar	FFHQ
			Att.ACC
<b>No Protection</b>	74.15	82.27	99.60
<b>Data Centric Defense</b>	74.12	85.71	0.00

Table 3.7: Defense performance against PPA on various datasets. Results are averaged over 6 randomly selected targets.

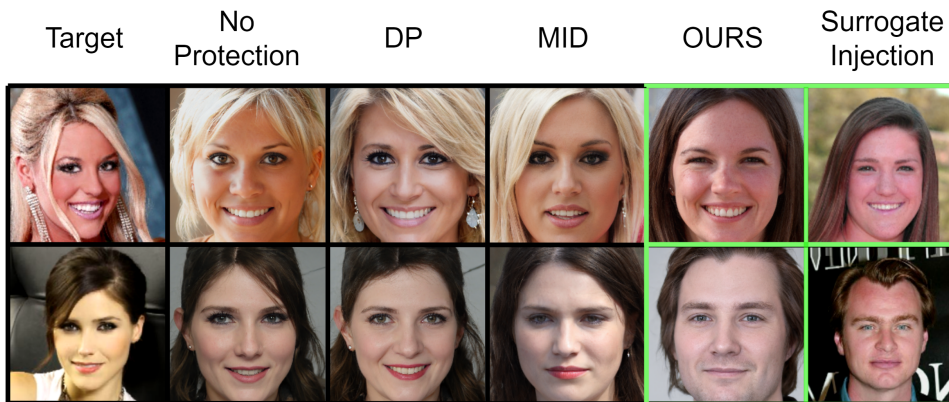
## 3.4 Discussion

### 3.4.1 Similarity between Target samples and Surrogate samples

Our experiments show that the surrogates that differ from the target samples aid the defense. Such a sample results in the reconstruction of images that are completely different from the target sample for e.g. using a female with black hair as a surrogate for a male with blonde hair results in the protection of blonde haired male.



(a) Comparison of the recovered images for various defenses with Data-Centric Defense. The recovered image is visually similar to the injected samples, thus protecting the target sample from the Model Inversion Attack



(b) Representation of the surrogate samples injected in the training set for Data-Centric Defense. The images in each row represent images belonging to the same identity with the true image on the left and an example of the surrogate injected on the right

Figure 3.3: Visual representation of recovered faces of Model Inversion Attacks



For the CelebA dataset, images are annotated by attributes like wearing a hat, wearing glasses, gender, hair color, etc. The most easily distinguishable attributes were chosen namely gender and to carry out our study. As shown in Table 3.8, two identities were utilized - one was a full match (male with black hair for a male with black hair), and the other was a full mismatch (female with blonde hair for a male with black hair). Except for a female with black hair which has a higher attack accuracy of 47.99%, with the other examples showcasing an attack accuracy of less than 10%. A full mismatch is the best, resulting in full protection of the data (attack accuracy of 0.00%) for three out of four cases.













Attribute		Defense Performance								
Gender	Hair Color	ACC	Att. ACC	ACC(--)	Att. ACC(--)	ACC(++)	Att. ACC(++)	ACC(++)	Att. ACC(++)	
Male	Black		83.33	96.77		81.67	0.00		100.00	5.99
Female	Black		100.00	100.00		100.00	4.99		100.00	47.99
Female	Blonde		85.71	92.00		81.14	0.00		85.71	7.99
Male	Blonde		75.00	100.00		69.00	0.00		75.00	0.00

Table 3.8: Defense performance with full mismatch and full match surrogate samples.

### 3.4.2 Non-zero diversity among surrogate samples within same class

Celebrity datasets are convenient for selecting surrogate samples owing to diversity and easy availability. The evaluation is carried out on three scenarios for the same amount of data: 1. No-Dup: Each surrogate image is unique. 2. Dup-5: Each target has 5 surrogate images duplicated, 3. Dup-1: A duplicated single surrogate sample. Furthermore, depending on the quality of the surrogate image, “Dup-1 High” for higher-quality images and “Dup-1 Low” for lower-quality images based on annotated features like hair color.

Table 3.9 shows that No-Dup has an attack accuracy of 4.5%. Dup-5 further deteriorates the attack accuracy to 0.80%. This shows that the diversity of surrogates does help in better defense. Our hypothesis based on our experiments suggests that the model performs better on high-quality images than on low-quality surrogates.

	<b>No Protection</b>	<b>Dup-1-Low</b>	<b>Dup-1-High</b>	<b>Dup-5</b>	<b>No-Dup</b>
<b>ACC-all</b>	86.95	86.92	86.24	86.97	86.57
<b>ACC-tar</b>	100.00	96.47	97.13	97.52	97.15
<b>Att. ACC</b>	100.00	22.50	18.00	0.80	4.50

Table 3.9: Impact of diversity and quality of surrogate samples within the same class.

# Chapter 4

## Membership Inference

Membership Inference Attacks solve a binary classification problem to identify the samples belonging to the training set of the target model. The target model has high confidence in the samples which the model has already seen. The proposed method aims to ensure correct classification with reduced confidence in the samples by replacing them with their augmented versions. Unlike regular data augmentation, this technique is used by the data owners to augment their data before sending it to the model owners for the downstream task.

### 4.1 Proposed Methodology

This research takes into account the scenario where the model owner, responsible for training the machine learning model (denoted as  $f_\theta$ ) is not the owner of the data and crawls one or multiple sources to gather useful information for the model training.

The objective of the data owner is to modify the input-label pair  $(x, y)$  by augmenting the inputs with a perturbation denoted as  $\delta$ . This modification replaces the original input, resulting in a new data sample of the form  $(x + \delta, y)$ , maintaining the target label. The generation of the augmented samples is inspired by [51] (pseudo-code given in B.2) and involves the following steps:

- Proxy model pre-training on a shared domain knowledge dataset
- Proxy model fine-tuning on the data of the data owner
- Replacement of original samples with augmented samples

### Data Owner Knowledge

In the given scenario, the data owner lacks specific knowledge about the target model’s architecture and other training-related parameters. However, data owners do possess information about the task for which the data will be utilized, such as image classification or face recognition. This general information is typically made public to gather insights from trusted external sources.

To address this limitation, a proxy model is employed for generating augmentations to protect sensitive data. This proxy model is trained on a public dataset that shares domain knowledge relevant to the task at hand. The pre-trained proxy model is further finetuned to serve as a substitute, enabling the generation of patches for sensitive data.

### Design of Augmented Samples

The injected augmentation ( $\delta$ ), is carefully selected in such a way that the model  $f_\theta$ , which is trained on the modified input  $x + \delta$ , relies on  $\delta$  to make its prediction. The injected augmentation patch, denoted as  $\delta$ , is carefully selected to ensure that the model  $f_\theta$ , which is trained on the modified input  $x + \delta$ , incorporates and utilizes  $\delta$  in its prediction process. Figure.4.1 shows what an augmented data sample for the given sample looks like.

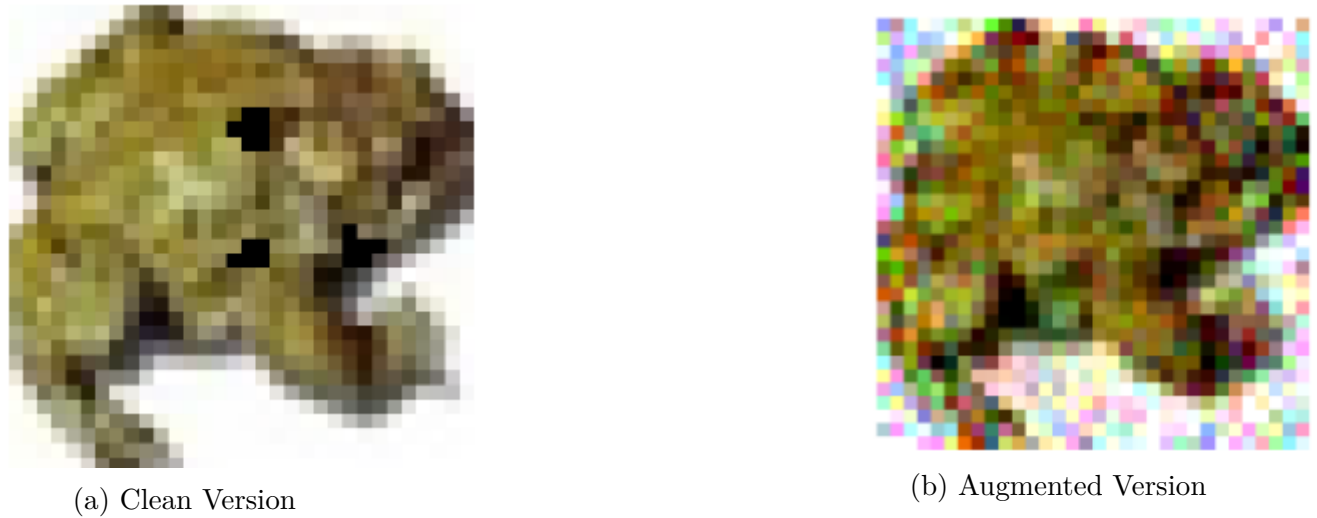


Figure 4.1: Clean and augmented version of images

### Generating augmented dataset

The augmentations are generated (shown in Figure 4.2) by leveraging the utilization of a pre-trained model ( $f_{\theta_{surr}}$ ), which possesses domain knowledge in the form of learned features. This model is trained on a dataset that shares the same domain knowledge as the training dataset, e.g. for a face recognition task, the model is trained on a publicly available face recognition dataset. The model thus becomes proficient at extracting general semantic features responsible for classifying different images from various classes for the downstream task.

The pre-trained model functions as a feature extractor which is then fine-tuned on the data owner's dataset, enabling the extraction of meaningful semantic features from the data owner's samples. This finetuned model is used to synthesize the augmentation using stochastic gradient descent. The model batches the samples and takes an average of the gradients over these samples.

The augmentation patch is then updated with the earlier calculated gradient value ensuring allowable design limit  $\Delta$  the  $l_p$  ball constraint ( $\delta : \|\delta\|_p \leq \epsilon$ ).

The  $\delta$  is designed to iteratively solve the optimization problem:

$$\delta^* = \arg \min_{\delta \in \Delta} \sum_{(x,y) \in \mathcal{D}} \mathcal{L}(f_{\theta_{surr}}(x + \delta), y) \quad (4.1)$$

These extracted features are then used to generate a  $\delta$  that resembles the target class. Bounding of  $\delta$  ensures that when the augmented image  $x + \delta$  is created, it remains visually similar to the original input  $x$ , as the changes introduced by  $\delta$  are limited within certain visual boundaries. Figure 4.3 shows how an unbounded  $\delta$  can cause the image to look visually dissimilar to the original image.

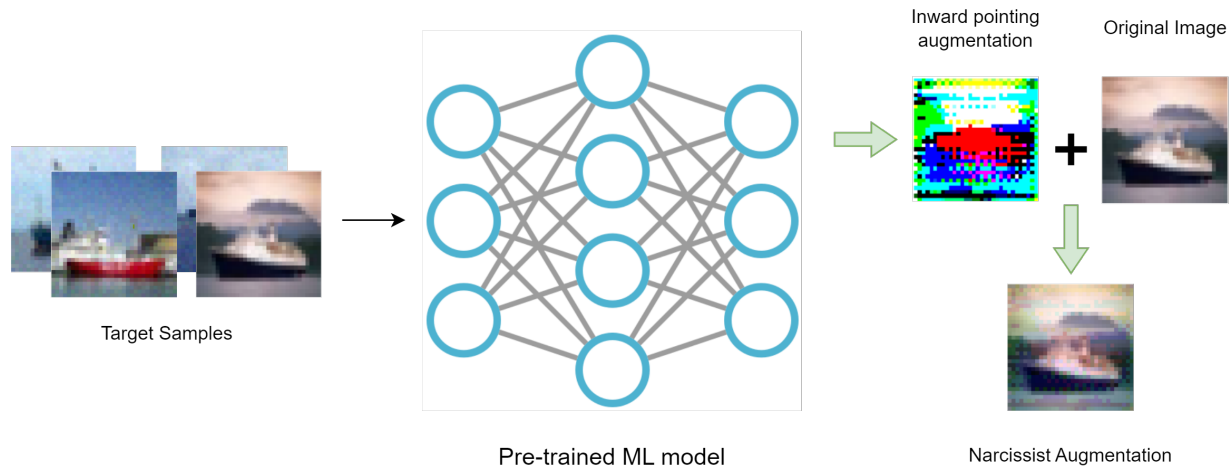


Figure 4.2: Pipeline to generate augmented samples( $x + \delta$ ) for a target class from a pre-trained model  $f_{\theta_{surr}}$  finetuned on the data belonging to the target class. The target samples are used to generate an ‘inward pointing’( $\delta$ ) patch by solving an optimization problem to minimize the loss on the augmented data samples, which classifies as the target sample.



Figure 4.3: Visual distortion of an image as the magnitude ( $l_p$ norm) of augmentation increases. The leftmost image is the original image followed by augmentations with the rightmost image being the one with the highest augmentation ( $l_p$ ).

## 4.2 Experiments

### 4.2.1 Attack algorithm

For a thorough study of the proposed data-centric defense against membership inference attacks, the proposed defense is evaluated on a variety of attack techniques, using one example of each of the above-mentioned attacks: 1. Classifier-based attack [38], 2. Metric-based attack (entropy, confidence, loss)[41, 49], 3. Likelihood Ratio Attack (LiRA) [7]. 4. Label-Only Attack [12, 26]

Half of the CIFAR-10 dataset (25000) makes up the train set and the remaining 25000, the test set. A random subset of 250 samples (1%) out of the train set with an almost equal representation from each of the 10 classes was selected as the data samples to be replaced by its augmented versions. For the defense evaluation over membership inference attacks, 250 original images selected for replacement and 250 samples from the test set were combined. Thus as a random guess results in an attack accuracy of 50%, the goal is to find augmentations that reduce the membership inference attack accuracy as close to the level of a random guess to provide perfect protection.

### 4.2.2 Baseline Augmentations for Comparison

Each of the baseline techniques affects the defense performance based on the change in the hyperparameter values. As these values change, it alters the privacy-utility tradeoff which is presented in Appendix C.2. The representation of the baseline augmentations is given in Figure 4.4.

**Gaussian Noise** These samples are generated with the mean kept as the clean version with varying  $\sigma$  ( $l_p$  ball) to alter the degree of visual similarity.

**Adversarial Noise** A proxy model is selected which is trained on Tint-ImageNet for 60 epochs. This pre-trained model is then finetuned for 5 epochs on our dataset. Following this, adversarial samples were generated using [28] by considering two hyperparameters step-size  $\alpha$  and  $\epsilon$  for  $l_p$  ball.

**Random Crop** Random Crop augments the  $32 * 32$  image by adding zero-padding around the image to create an image of a larger size. Following this, an image of the size  $32 * 32$  was cropped, which is then supplied to the model. The padding is chosen based on the best privacy-utility tradeoff.

**CutOut** In CutOut[13], a random  $n * n$  pixel area is masked from the given image. The masked image is then used to train the model.



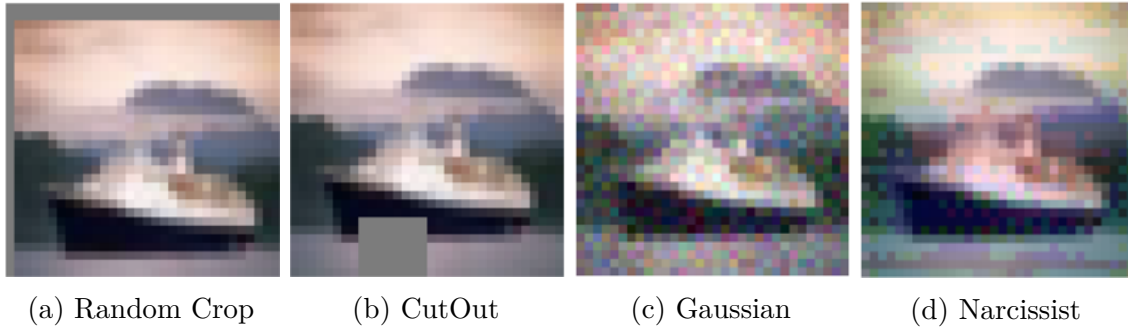


Figure 4.4: Visual Representation of various baseline augmentations to the target image

### 4.2.3 Evaluation Setup - Attacks, Datasets, and Models

The evaluation was done on a variety of attack types and models. The dataset used was CIFAR-10. There are two sets of models used, one is the augmentation generation model and the other is the target model. An overview is presented in Table 4.1 with details in Appendix A.

Attack Type	Dataset	Model	Pre-Trained Model
Binary Classifier		ResNet-18	
Loss-Based		ResNet-18	
Confidence-Based	CIFAR-10	ResNet-18	ResNet-18
Entropy-Based		ResNet-18	
LiRA		WideResNet-28-2	
Label-Only		CNN	

Table 4.1: Overview of the attack methods, datasets, and models on which the data-centric method is evaluated.

The pre-trained model used is ResNet-18 to generate the augmentation  $\delta$ , as for the target model, different were used as target models like ResNet-18 for binary classifier-based attacks and metric-based attacks, a WideResNet-28,2[50] for LiRA, and CNN model for the label-only attack as open-sourced by the authors [7, 26]. A detailed description of the models and datasets is in Appendix A.

## 4.3 Results and Discussions

The membership inference attacks were evaluated by considering different augmentations techniques to study how these augmentations affect the model performance and membership inference. The augmentation providing a better privacy-utility tradeoff is the one that has a lower AUC and TPR at low FPR while having a high ACC-tar.

### Evaluation Metrics

**AUC-ROC** The Receiver Operating Characteristic (ROC) Curve(Figure 4.5) presents a tradeoff between a True Positive Rate(TPR) and a False Positive Rate(FPR) at various threshold settings. This is a better identifier for the binary classification problem of membership. The Area Under Curve (AUC) is the ability to distinguish between the samples for membership. An AUC of 0.5 represents a random guess. i.e. zero ability to correctly identify a member sample. For a perfect defense, the AUC should be as close to 0.5 as possible.

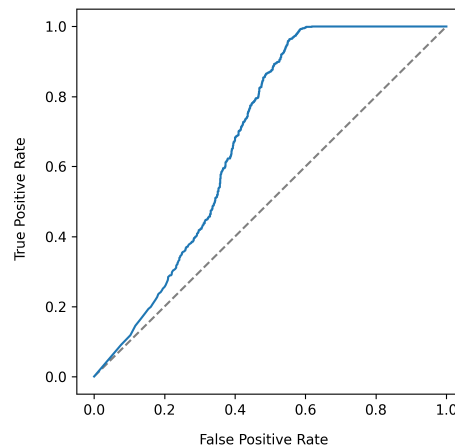


Figure 4.5: The ROC curve

**TPR at low FPR** [7] suggests a modified version of the AUC-ROC for evaluating the potency of membership inference attacks. [7] shows the ROC curve on a logarithmic scale while reporting the results as TPR at a fixed FPR (0.1% or 0.001%). Figure 4.6 shows the ROC curve along the logarithmic scale which is used to calculate TPR at low FPR. An ideal defense should have a TPR value of 0.0000 for low FPR.

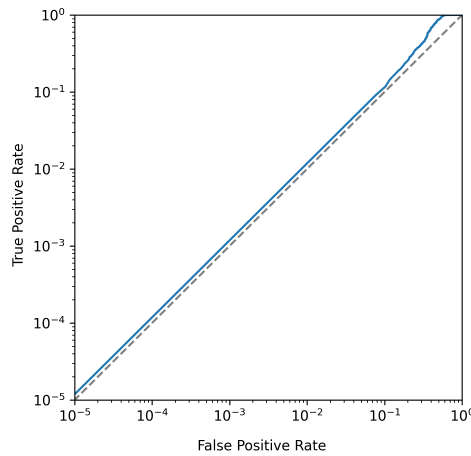


Figure 4.6: The logarithmic scale ROC curve to find TPR at a low FPR

**ACC-tar** Classification accuracy over the target samples which have been replaced by their augmented versions, represented in %. The aim is to replace the samples while ensuring that the model is still able to correctly classify them to their respective classes. Thus, ideally, the value for ACC-tar should be as high as possible.

### Metric-Based Attacks

Tables 4.2, 4.3 and 4.4 show the results for how augmentations affect metric-based attacks. As observed, the AUC for membership inference attacks on clean samples stands at 0.7120, 0.7151, and 0.7151 for confidence, modified entropy, and loss respectively.

Gaussian noise proves to be slightly effective in protecting the data samples with the AUC decreasing marginally to 0.5804, 0.5890, and 0.5865 for respective metric-based attacks while having an ACC-tar of 89.00%. Adversarial noise proves to enhance membership inference increasing the TPR at a low FPR. As the results indicate, these attacks are not potent enough when it comes to identifying the membership status of individual samples with confidence with the TPR at a low FPR being 0.0000 for all augmentations for all the attacks except for modified entropy. For modified entropy, the TPR increases from 0.0064 for the clean model to 0.0120 for Gaussian models and 0.0128 for Adversarial models. The values for Cutout are lower than for clean models for all the metrics. Random Crop reported an AUC of 0.5660, 0.5680, and 0.5672 at a lower ACC-tar of 87.20. Narcissist Augmentation provides a better privacy-utility tradeoff with AUC scores of 0.5573, 0.5685, and 0.5675 for respective attacks while having an ACC-tar of 90.00%.

	Clean	Gaussian Augmentation	Adversarial Augmentation	Random Crop	CutOut	Narcissist Augmentation
<b>AUC</b>	0.7120	0.5804	0.5752	0.5660	0.6738	0.5573
<b>TPR@0.1% FPR</b>	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
<b>ACC-tar</b>	100.00	89.00	87.70	87.20	97.60	90.00

Table 4.2: Overview of the augmentation methods for Metric Based Attacks - Confidence

	Clean	Gaussian Augmentation	Adversarial Augmentation	Random Crop	CutOut	Narcissist Augmentation
<b>AUC</b>	0.7151	0.5890	0.5835	0.5680	0.6802	0.5685
<b>TPR@0.1% FPR</b>	0.0064	0.0120	0.0240	0.0040	0.0080	0.0064
<b>ACC-tar</b>	100.00	89.00	87.70	87.20	97.60	90.00

Table 4.3: Overview of the augmentation methods for Metric Based Attacks - Modified Entropy

	Clean	Gaussian Augmentation	Adversarial Augmentation	Random Crop	CutOut	Narcissist Augmentation
<b>AUC</b>	0.7151	0.5865	0.5813	0.5672	0.6800	0.5675
<b>TPR@0.1% FPR</b>	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
<b>ACC-tar</b>	100.00	89.00	87.70	87.20	97.60	90.00

Table 4.4: Overview of the augmentation methods for Metric Based Attacks - Loss

### Binary Classifier Attacks

As the data samples are augmented (shown in Table 4.5) the AUC from 0.7151 for clean samples decreases to 0.5684 for Gaussian noise with an accuracy of 89% and 0.6544 CutOut with 97.60%, a bigger drop is seen for the random crop with AUC 0.5463 and accuracy 87.20% and Adversarial samples with AUC of 0.5500 and accuracy of 87.70%. Narcissist Augmentation lowers the AUC to 0.5441 while having a higher accuracy of 90.00%.

	Clean	Gaussian Augmentation	Adversarial Augmentation	Random Crop	CutOut	Narcissist Augmentation
<b>AUC</b>	0.7151	0.5684	0.5500	0.5463	0.6544	0.5441
<b>TPR@0.1% FPR</b>	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
<b>ACC-tar</b>	100.00	89.00	87.70	87.20	97.60	90.00

Table 4.5: Overview of the augmentation methods for Binary Classifier Attacks

### Label-Only Attacks

Evaluation against Label-Only Attacks is seen in Table 4.6, with AUC and TPR 0.6689 and 0.0043 for Clean samples, 0.6578 and 0.0040 for Gaussian Augmentation, 0.5780 and 0.0040 for random crop whereas it is 0.5692 and 0.0040 for Narcissist Augmentation. The target accuracy is 100%, 97%, 91%, and 86% for Clean, Gaussian, Random Crop, and Narcissist Augmentation.

	Clean	Gaussian Augmentation	Random Crop	Narcissist Augmentation
<b>AUC</b>	0.6689	0.6578	0.5780	0.5692
<b>TPR@0.1% FPR</b>	0.0043	0.0040	0.0040	0.0040
<b>ACC-tar</b>	100.00	97.00	91.00	86.00

Table 4.6: Overview of the augmentation methods for Label-Only Attacks

### Likelihood Ratio Attacks

As seen in Table 4.7, the TPR at low FPR is quite significant having values 0.0470, 0.0690, and 0.0800 for Clean, Gaussian augmentation, and adversarial samples respectively whereas it drops to 0.0005 for random crop and rises to 0.0135 for CutOut. The AUC starting at 0.8046 for clean samples decreases to 0.7824 for Gaussian noise and 0.5920 for Random Crop whereas, for Adversarial Noise and Cutout, it reaches higher to 0.8252 and 0.7644 respectively. For each of the augmentations, the target accuracy is 90.10%, 85.40%, 90.08%, 84.88%, and 89.00%. For Narcissist Augmentation the AUC is 0.5540 and TPR of 0.0050 with the target accuracy of 83.78%.

	Clean	Gaussian Augmentation	Adversarial Augmentation	Random Crop	CutOut	Narcissist Augmentation
<b>AUC</b>	0.8046	0.7824	0.8252	0.5920	0.7644	0.5540
<b>TPR@0.1% FPR</b>	0.0470	0.0690	0.0800	0.0005	0.0135	0.0050
<b>ACC-tar</b>	90.10	85.40	90.08	84.88	89.00	83.78

Table 4.7: Overview of the augmentation methods for LiRA

A detailed explanation of the effects of change of hyperparameters on the privacy-utility tradeoff can be seen in Appendix C.2. The results show that using adversarial samples does more harm to the model when the perturbation is low. This conforms with the findings of [43]. [43] displays the ill effects of robust model training suggesting that adversarially trained models lead to increased privacy risks. As the adversarial samples are not mislabeled, they affect the decision boundary of the model leading to increased exposure of clean data.

Adding Gaussian augmentation, however, adds confusion to the model as it masks some of the semantic features protecting the original samples. With random noise, the augmented sample preserves the information from the original semantic features. Narcissist Augmentation on the other hand augments the samples such that the augmented sample lies even more inside the decision boundary. With the original sample being slightly pushed closer to the decision boundary, the target model has less confidence in the original sample, leading to its protection from membership inference attacks.

While the results indicate that a better privacy-utility tradeoff within the existing is given by random cropping. However, the performance takes a hit as the padding values increase beyond four. The intuition is that as a result of spurious correlation, the model focuses more on the background semantics than the foreground semantics to make the label decision. [31] supports this argument indicating that background distortion deteriorates the decision confidence more than foreground distortions.

# Chapter 5

## Conclusion and Future Work

The thesis proposes a data protection mechanism that is driven by user preferences and allows for individual control. The approach, known as Data-Centric Defense, guarantees the preservation of data privacy without compromising the utility or accuracy of the model’s classifications. A thorough evaluation demonstrated that the proposed defense surpasses existing model-centric defenses in terms of both privacy and utility when it comes to model inversion attacks.

One limitation is that it results in a fourfold increase in the number of examples associated with the target class in the context of model inversion attacks. Malicious model trainers can identify and exclude surrogate samples that exhibit visual dissimilarities from the target class, thus compromising the effectiveness of data protection. To address this issue, future endeavors should focus on obscuring the injected samples to deceive human inspection, thereby mitigating these limitations.

The theory of Data Augmentation has been extensively surveyed in the setting of mitigating membership inference attacks, primarily aiming to address model overfitting concerns. While these augmentations contribute to enhancing the overall generalization of the model, they may not yield remarkable results when applied to unprotected versions of the sample.



Besides, the model owner is responsible for applying these methods leaving the data owner with lesser control over one's own data. This thesis expands upon the notion of a data-centric defense and by applying it to membership inference attacks. By doing so, an improved trade-offs between privacy and utility compared to certain existing data augmentation techniques is showcased.

The future work should aim to discover an improved data augmentation approach specifically designed for membership inference attacks to provide a better privacy-utility tradeoff. The objective must be to develop a technique that effectively safeguards the benign samples, ensuring their privacy, while simultaneously preserving a high level of classification accuracy.

# Bibliography

- [1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318, 2016.
- [2] 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*, 2022.
- [3] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017.
- [4] Dimitri P Bertsekas. Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334, 1997.
- [5] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)*, pages 67–74. IEEE, 2018.
- [6] Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *USENIX Security Symposium*, volume 267, 2019.
- [7] Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian

- Tramer. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 1897–1914. IEEE, 2022.
- [8] Dingfan Chen, Ning Yu, and Mario Fritz. Relaxloss: defending membership inference attacks without losing utility. *arXiv preprint arXiv:2207.05801*, 2022.
- [9] Si Chen, Feiyang Kang, Nikhil Abhyankar, Ming Jin, and Ruoxi Jia. Data-centric defense: Shaping loss landscape with augmentations to counter model inversion. 2023. Under Review.
- [10] Yufei Chen, Chao Shen, Yun Shen, Cong Wang, and Yang Zhang. Amplifying membership exposure via data poisoning. *arXiv preprint arXiv:2211.00463*, 2022.
- [11] 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 workshops*, pages 1924–1932, 2017.
- [12] Christopher A. Choquette-Choo, Florian Tramer, Nicholas Carlini, and Nicolas Papernot. Label-only membership inference attacks. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 1964–1974. PMLR, 18–24 Jul 2021.
- [13] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.
- [14] 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 *23rd {USENIX} Security Symposium ({USENIX} Security 14)*, pages 17–32, 2014.

- [15] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.
- [16] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.
- [17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [18] Nils Homer, Szabolcs Szelinger, Margot Redman, David Duggan, Waibhav Tembe, Jill Muehling, John V Pearson, Dietrich A Stephan, Stanley F Nelson, and David W Craig. Resolving individuals contributing trace amounts of dna to highly complex mixtures using high-density snp genotyping microarrays. *PLoS genetics*, 4(8):e1000167, 2008.
- [19] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.
- [20] Jinyuan Jia, Ahmed Salem, Michael Backes, Yang Zhang, and Neil Zhenqiang Gong. 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, 2019.
- [21] 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*, pages 15045–15053, 2022.
- [22] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for gen-

- erative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.
- [23] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *Advances in Neural Information Processing Systems*, 33:12104–12114, 2020.
- [24] Yigitcan Kaya and Tudor Dumitras. When does data augmentation help with membership inference attacks? In *International conference on machine learning*, pages 5345–5355. PMLR, 2021.
- [25] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Fei-Fei Li. Novel dataset for fine-grained image categorization: Stanford dogs. In *Proc. CVPR workshop on fine-grained visual categorization (FGVC)*, volume 2. Citeseer, 2011.
- [26] Zheng Li and Yang Zhang. Membership leakage in label-only exposures. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 880–895, 2021.
- [27] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.
- [28] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.
- [29] Malgorzata Magdziarczyk. Right to be forgotten in light of regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such

- data, and repealing directive 95/46/ec. In *6th International Multidisciplinary Scientific Conference on Social Sciences and Art Sgem 2019*, pages 177–184, 2019.
- [30] Tom Michael Mitchell et al. *Machine learning*, volume 1. McGraw-hill New York, 2007.
- [31] Mazda Moayeri, Phillip Pope, Yogesh Balaji, and Soheil Feizi. A comprehensive study of image classification model sensitivity to foregrounds, backgrounds, and visual attributes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19087–19097, 2022.
- [32] Milad Nasr, Reza Shokri, and Amir Houmansadr. Machine learning with membership privacy using adversarial regularization. In *Proceedings of the 2018 ACM SIGSAC conference on computer and communications security*, pages 634–646, 2018.
- [33] Hong-Wei Ng and Stefan Winkler. A data-driven approach to cleaning large face datasets. In *2014 IEEE international conference on image processing (ICIP)*, pages 343–347. IEEE, 2014.
- [34] Stuart L Pardo. The california consumer privacy act: Towards a european-style privacy regime in the united states. *J. Tech. L. & Pol’y*, 23:68, 2018.
- [35] Maria Rigaki and Sebastian Garcia. A survey of privacy attacks in machine learning. *arXiv preprint arXiv:2007.07646*, 2020.
- [36] Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, Yann Ollivier, and Hervé Jégou. White-box vs black-box: Bayes optimal strategies for membership inference. In *International Conference on Machine Learning*, pages 5558–5567. PMLR, 2019.
- [37] Ahmed Salem, Yang Zhang, Mathias Humbert, Pascal Berrang, Mario Fritz, and Michael Backes. Ml-leaks: Model and data independent membership inference attacks and defenses on machine learning models. *arXiv preprint arXiv:1806.01246*, 2018.

- [38] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pages 3–18. IEEE, 2017.
- [39] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [40] Congzheng Song, Thomas Ristenpart, and Vitaly Shmatikov. Machine learning models that remember too much. In *Proceedings of the 2017 ACM SIGSAC Conference on computer and communications security*, pages 587–601, 2017.
- [41] Liwei Song and Prateek Mittal. Systematic evaluation of privacy risks of machine learning models. In *USENIX Security Symposium*, volume 1, page 4, 2021.
- [42] Liwei Song, Reza Shokri, and Prateek Mittal. Membership inference attacks against adversarially robust deep learning models. In *2019 IEEE Security and Privacy Workshops (SPW)*, pages 50–56. IEEE, 2019.
- [43] Liwei Song, Reza Shokri, and Prateek Mittal. Privacy risks of securing machine learning models against adversarial examples. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 241–257, 2019.
- [44] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The german traffic sign recognition benchmark: a multi-class classification competition. In *The 2011 international joint conference on neural networks*, pages 1453–1460. IEEE, 2011.
- [45] Lukas Struppek, Dominik Hintersdorf, Antonio De Almeida Correia, Antonia Adler, and Kristian Kersting. Plug & play attacks: Towards robust and flexible model inversion attacks. In *International Conference on Machine Learning*, pages 20522–20545. PMLR, 2022.

- [46] Florian Tramèr, Reza Shokri, Ayrton San Joaquin, Hoang Le, Matthew Jagielski, Sanghyun Hong, and Nicholas Carlini. Truth serum: Poisoning machine learning models to reveal their secrets. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 2779–2792, 2022.
- [47] Tianhao Wang, Yuheng Zhang, and Ruoxi Jia. Improving robustness to model inversion attacks via mutual information regularization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 11666–11673, 2021.
- [48] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.
- [49] Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st computer security foundations symposium (CSF)*, pages 268–282. IEEE, 2018.
- [50] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.
- [51] Yi Zeng, Minzhou Pan, Hoang Anh Just, Lingjuan Lyu, Meikang Qiu, and Ruoxi Jia. Narcissus: A practical clean-label backdoor attack with limited information. *arXiv preprint arXiv:2204.05255*, 2022.
- [52] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.
- [53] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Haibin Lin, Zhi Zhang, Yue



- Sun, Tong He, Jonas Mueller, R. Manmatha, Mu Li, and Alexander Smola. Resnest: Split-attention networks, 2020.
- [54] 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*, pages 253–261, 2020.

# Appendices

# Appendix A

## Datasets and Models

### A.1 Datasets

**CelebA** A 202,599 strong face dataset consisting of 10,177 celebrities with each image of the size 178x218. The images are resized to 224x224 and cropped by a face factor of 0.65. Our dataset contains 27,034 training images and 3004 test images belonging to the 1000 most frequent samples in CelebA.

**FaceScrub** The FaceScrub dataset is a substantial collection of face images, consisting of 106,863 images. It includes 530 celebrities, with an equal split of 265 males and 265 females. Each celebrity has approximately 200 images associated with them. To simplify the dataset, we assigned integer labels 0-264 to male celebrities and 265-529 to female celebrities. Following the guidelines established in PPA, we utilized 34,090 images for training and 3,788 images for testing purposes.

**Stanford Dogs** The Stanford Dogs dataset is a collection of images used for dog classification, featuring 120 different dog breeds. The dataset consists of 18,522 training samples and 2,058 test samples, totaling 20,580 images. The images in the dataset exhibit variations in size, style, and content. Some images even include multiple dog breeds.

**GTSRB** The German Traffic Sign Recognition Benchmark (GTSRB) dataset is specifically designed for traffic signal recognition. It consists of 35,288 training images and 12,630 test images, all categorized into 43 different classes representing distinct traffic signs. To maintain consistency, all the images in the dataset have been resized to dimensions of 32x32 pixels.

**VGGFace2** The VGGFace2 dataset is a face recognition dataset that focuses on large-scale recognition tasks. The images in this dataset were sourced from Google Image Search, resulting in a wide range of variations in terms of pose, age, illumination, ethnicity, and profession. However, it's important to note that the dataset link is no longer active on the official website, limiting the availability of images. As a result, we were only able to gather 1984 training images and 416 test images, distributed across 8 distinct classes, for this dataset.

**FFHQ** The FFHQ dataset is known for its exceptional quality and diversity, surpassing both the CelebA and FaceScrub datasets. It consists of a remarkable collection of 70,000 face images, each having a resolution of 1024x1024.

**MetFaces** The image dataset in question contains 1,336 unique images, each depicting artistic renditions of human faces. These images showcase a wide range of artistic variations. However, it's important to note that the dataset is biased and lacks adequate representation of individuals with darker skin tones.

**Animal Faces-HQ (AFHQ)** The dataset consists of 16,130 images of wildlife animals, cats, and dogs, all sized 512x512 pixels. However, for the specific purpose of evaluating the Stanford Dogs dataset, only the images of dogs are selected and used.

**TSRD** The collection comprises 58 different categories, encompassing a total of 6,164 traffic sign images. These images are further divided into training and test sets, with 4,170 images allocated for training and 1,994 images for testing.

**CIFAR-10** CIFAR-10 is a dataset consisting of low-resolution (32x32) color images belonging to 10 classes. It consists of 60,000 images with 50,000 training images and 10,000 test images respectively, split equally into 10 classes.

**TinyImageNet** As the name suggests it is a miniaturized version of the full ImageNet dataset. The colored dataset has 200 classes each containing about 500 images totalling up to 100,000 images in all. A downsized version of ImageNet has images of size 64x64, it includes 500 training images, 50 validation images, and 50 test images.

## A.2 Models

The models used in our experiments are VGG-16, ResNet-152, ResNeSt-101, ResNext-101, ResNet-18, WideResNet-10-2, CNN, and DenseNet-169.

**VGG** The VGG model, developed by the Visual Geometry Group at the University of Oxford, is a widely recognized deep convolutional neural network architecture utilized for image classification tasks. The model VGG-16 comprises up to 16 different types of layers.

**ResNet** ResNet tackles the difficulty of training deep neural networks by incorporating residual connections. These connections enable the flow of information, mitigating the issue of vanishing gradients. ResNet-18 and ResNet-152 are specific alternatives, each with a

different depth configuration.

**ResNeSt** ResNeSt is an advanced variant of ResNet that enhances deep learning models with its "Split-Attention" concept.

**ResNext** ResNext enhances the ResNet architecture through the introduction of "cardinality." This concept involves parallel branches within each residual block, enabling the model to capture diverse and detailed features effectively.

**DenseNet** Introduces dense connections, where each layer is directly connected to every other layer in a feed-forward manner. The DenseNet-169 has 169 such layers having strong connections.

**WideResNet** Wide ResNet is a modification of ResNet that focuses on increasing the number of channels within each layer, enhancing the model's representational capacity.

**CNN** Convolutional Neural Networks (CNNs) are specialized deep learning models primarily utilized for computer vision applications. Typically comprises multiple convolutional, pooling, and fully connected layers.

# Appendix B

## Pseudo-Code of the proposed algorithms

### B.1 Model Inversion

---

**Algorithm 1** Algorithm of Data-Centric Defense for Model Inversion Attacks

---

**Input:** Entire label set  $\mathcal{Y}$ , target label set  $S_{tgt}$ , raw training samples corresponding to the target label set  $D_{tgt-raw}$ , mislabel ratio  $\pi_1$  and  $\pi_2$ , noise magnitude  $\epsilon_1$  and  $\epsilon_2$ .

Denote samples from class  $y_i$  as  $\{(x_{ij}, y_i) : j = 1, \dots, m_i\}$ , where  $m_i$  is the number of samples of this class.

**for**  $i \in S_{tgt}$  **do**

1. Find a surrogate class not present in  $\mathcal{Y}$  and gather the same number of samples as class  $y_i$ . Relabel the gathered samples as class  $y_i$

$$D_i^1 = \{(x_{ij}^1, y_i) : j = 1, \dots, m_i\}.$$

2. Mislabel a small portion of raw target training samples with a ratio  $\pi_1$  using a random wrong label  $y' \sim \text{Uniform}(\mathcal{Y} \setminus y_i)$  to these samples

$$D_i^0 = \{(x_{ij}^0, y'_j) : j = 1, \dots, \lceil m_i \pi_1 \rceil\} \cup \{(x_{ij}^0, y_i) : j = \lceil m_i \pi_1 \rceil + 1, \dots, m_i\}.$$

3. Augment surrogate samples with Gaussian noise

$$D_i^2 = \{(x_{ij}^1 + \mu_j, y_i) : j = 1, \dots, m_i\}, \text{ where } \mu_j \sim \mathcal{N}(0, \epsilon_1^2).$$

4. Augment target samples with Gaussian noise and mislabel a portion of augmentations with ratio  $\pi_2$  using a random wrong label  $\tilde{y}$ :

$$D_i^3 = \{(x_{ij}^0 + \mu'_j, \tilde{y}_j) : j = 1, \dots, \lceil m_i \pi_2 \rceil\} \cup \{(x_{ij}^0 + \mu'_j, y_i) : j = \lceil m_i \pi_2 \rceil + 1, \dots, m_i\},$$

where  $\mu'_j \sim \mathcal{N}(0, \epsilon_2^2)$

**end for**

**return**  $\{D_i^0 \cup D_i^1 \cup D_i^2 \cup D_i^3 : i \in S_{tgt}\}$

---

## B.2 Memebership Inference

---

**Algorithm 2** Algorithm of Data-Centric Defense for Membership Inference Attacks[51]

---

**Input:**

$f_{\theta_{surr}}$  : Proxy Model

$D_t$  : Target Samples

$\epsilon$  : Permissible limit( $l_p$  ball radius) for the augmentation

$\delta_0 \leftarrow 0^{1*\text{dims of image}}$

**for**  $i \in (1, itr)$  **do**

1. Update the patch

$\delta_{i+1} \leftarrow \delta_i - \alpha \sum_{(x,t) \in \mathcal{D}} \mathcal{L}(f_{\theta_{surr}}(x + \delta), t)$

2. Apply the  $l_p$  ball constraint

$\delta_{i+1} \leftarrow \|\delta_{i+1}\|_p \leq \epsilon$

**end for**

**return**  $\delta_{itr}$

---



# Appendix C

## Hyperparameters

### C.1 Model Inversion

DP-SGD involves adding noise to the gradient followed by gradient clipping. The hyperparameters include the probability upper bound, denoted as  $\delta$ , which represents the likelihood of the model failing to provide privacy guarantees (roughly  $\frac{1}{\text{size of the dataset}}$ ), and the noise multiplier, denoted as  $\sigma$ , which is adjusted to achieve the desired privacy budget  $\epsilon$ . The learning rate and batch size remain fixed at the values used for normal model training, while the threshold for gradient clipping is set to a constant value of 1.

The goal of MID is to restrict the information conveyed by the model’s prediction about the input. To achieve this, MID introduces a hyperparameter denoted as  $\beta$ , which represents the weight assigned to the information loss that reduces the correlation between the output logit and the input. Detailed information is provided in Table C.1.

Table C.1: Privacy Parameters in DP-SGD and MID.

Attack Method	MID	DP		
	$\beta$	$\sigma$	$\delta$	$C$
<b>GMI</b>	0.2	1.0	$1e - 4$	1.0
<b>PPA</b>	0.07	0.1	$4e - 5$	1.0
<b>MIRROR</b>	0.003	2.0	$5e - 4$	1.0
<b>BREP-MI</b>	0.002	0.1	$4e - 5$	1.0

## C.2 Membership Inference

We have considered  $\sigma$  as the parameter for the  $l_p$  ball radius for Adversarial augmentation and Gaussian augmentation. The  $\sigma$  is progressively increased ranging from  $\frac{8}{255}$  up to  $\frac{128}{255}$  increasing by a factor of 2 at each stage. The step size  $\alpha$  is kept constant at 0.2 along with the number of epochs being 20 to construct Adversarial samples. For random cropping, we consider the amount of padding ( $p$ ) ranging from 1 to 8.

Figures C.1,C.2,C.3,C.4 showcase how different the hyperparameter choice affects the privacy-utility tradeoff. The points in the top-left corner indicate augmentations that have a high utility and also expose data privacy, whereas the points on the bottom right are those with low utility and high privacy. Ideally, for a good defense, the points should have a high utility and high privacy i.e. the points should lie in the top right-hand corner of the plot. The further away a point is from the axis, the better the privacy-utility tradeoff.

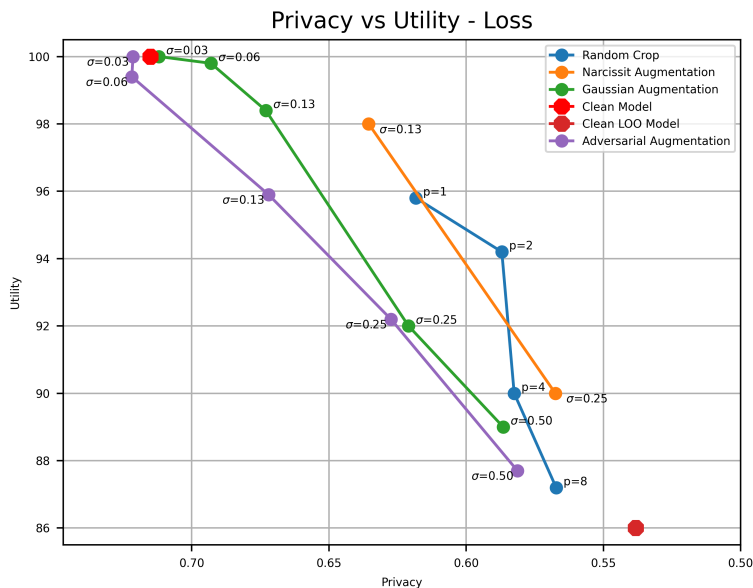


Figure C.1: Privacy - Utility curve for the loss based attacks

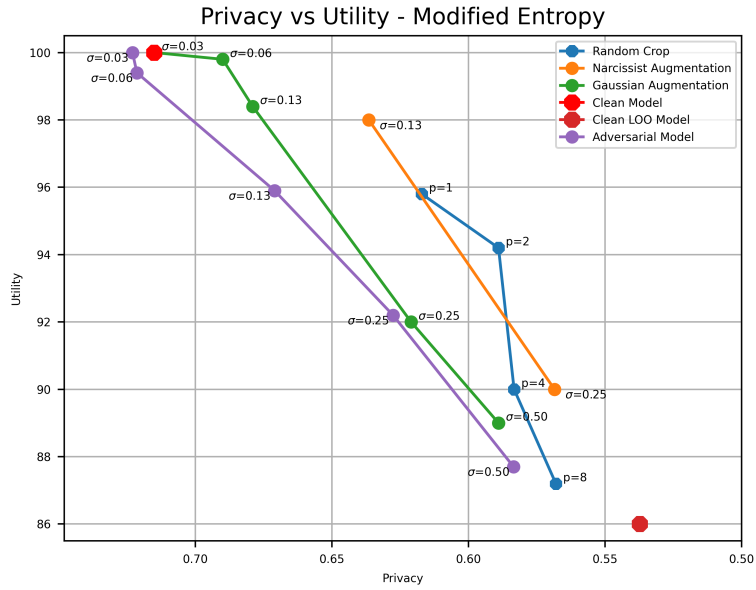


Figure C.2: Privacy - Utility curve for the modified entropy based attacks

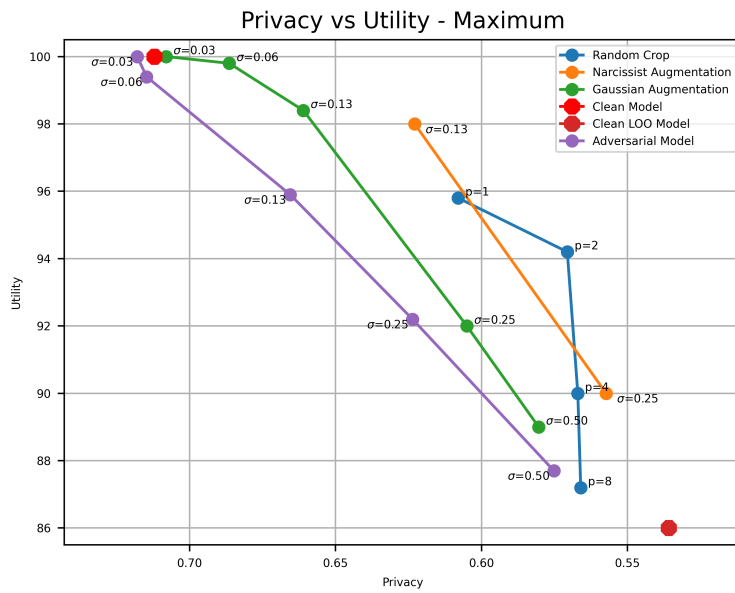


Figure C.3: Privacy - Utility curve for the maximum confidence based attacks

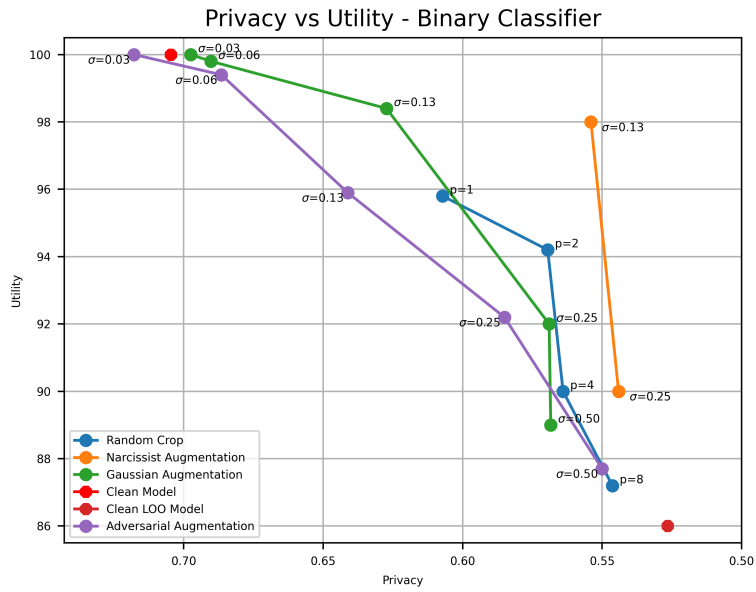


Figure C.4: Privacy - Utility curve for the binary classifier based attacks