Assessing the Impact of Image Dataset Features on Privacy-Preserving Machine Learning

Lucas Lange ●[1], Maurice-Maximilian Heykeroth[1], and Erhard Rahm ●[1]

Abstract: Machine Learning (ML) is crucial in many sectors, including computer vision. However, ML models trained on sensitive data face security challenges, as they can be attacked and leak information. Privacy-Preserving Machine Learning (PPML) addresses this by using Differential Privacy (DP) to balance utility and privacy. This study identifies image dataset characteristics that affect the utility and vulnerability of private and non-private Convolutional Neural Network (CNN) models. Through analyzing multiple datasets and privacy budgets, we find that imbalanced datasets increase vulnerability in minority classes, but DP mitigates this issue. Datasets with fewer classes improve both model utility and privacy, while high entropy or low Fisher Discriminant Ratio (FDR) datasets deteriorate the utility-privacy trade-off. These insights offer valuable guidance for practitioners and researchers in estimating and optimizing the utility-privacy trade-off in image datasets, helping to inform data and privacy modifications for better outcomes based on dataset characteristics.

Keywords: Data privacy, Image data, Privacy-preserving machine learning, Differential privacy

1 Introduction

Since the success of applications like ChatGPT[2] and DALL-E 2[3], new Artificial Intelligence (AI) technologies and projects emerge daily. These AI technologies are essentially improved Machine Learning (ML) models fine-tuned for specific tasks, learning rules and patterns from data to perform tasks like image classification. Large models like ChatGPT are often trained on internet data, submitted by humans, raising the question: if ML models are trained with personal data, can they leak information about individuals? Researchers have shown that ML models can be attacked. Fig. 1 shows one such attack, reconstructing training images from gradient information [Ge20]. These attacks can affect privacy and lead to potentially harmful consequences, if an attacker obtains a victim's sensitive personal information. Data breaches also generally violate GDPR [EC16] and may result in high fines, as seen with Cosmote Mobile Telecommunications, which paid €6,000,000 after a data breach [Eu22].

Privacy-Preserving Machine Learning (PPML) aims to prevent ML models from leaking sensitive information by increasing the effective application of privacy guarantees [XBJ21]. A common approach is introducing Differential Privacy (DP) to non-private ML models,

(a) Original training image. (b) Reconstructed image.
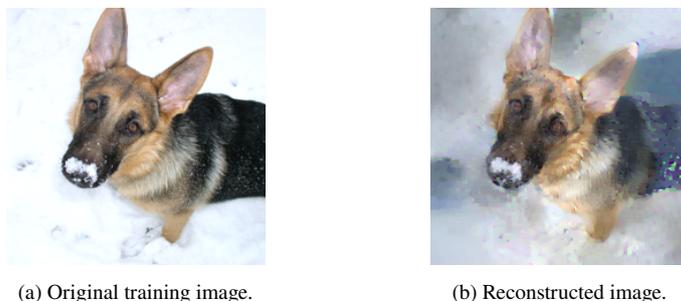
Fig. 1: Result of a model inversion attack in a federated learning scenario using gradient information. Left image shows the original image that was used to train the model, the right image shows the reconstructed image from an inversion attack. Results of the reconstruction attack by [Ge20].

turning them into private ones [AC19]. The main drawback is the trade-off between model utility and privacy guarantees, requiring significant system tweaking to balance both interests. Research has explored private learning approaches and PPML, but less is known about the impact of dataset characteristics on ML attacks or defenses [XBJ21]. Understanding these effects can accelerate efforts by guiding system tuning [Ca22; La23; Sh17; Tr21].

We investigate the influence of different dataset characteristics on the behavior of private and non-private image classification models. We measure this by training models on various differing datasets and attacking them with a state-of-the-art Membership Inference Attack (MIA), which infers whether a data sample was part of the model's training set. We focus on Convolutional Neural Network (CNN)-based image classification models, a common Machine Learning as a Service (MLaaS), since previous PPML research also underlines that image datasets and CNNs are prone to attacks [Ca22; Sh17; SP23; Tr21].

We extend previous work that have looked at the relation of dataset characteristics and MLs, by considering new metrics and private learning scenarios. This can help assess whether applying DP is worthwhile for a given dataset and what data considerations are needed for secure yet useful ML applications. We thereby take a more holistic view of the impact of dataset characteristics on both model utility and vulnerability to provide data-centric best practices for building private ML models. We discuss our final results and derived recommendations in Sect. 5.3. Our key findings resolve around an a priori analysis of vulnerability based on our dataset metrics. We further find that our models effectively mitigate most of the MIA threat with just a modest DP guarantee, achieving a more practical utility-privacy trade-off at low risk.

This work is structured continuing in Sect. 2, which presents essential concepts. Sect. 3 then reviews related work, while Sect. 4 details our experiments, with Sect. 5 discussing their results. Sect. 6 finally summarizes contributions and suggests future research directions.

2 Background

This section provides an understanding of essential methods relevant to the experiments.

2.1 Creating Private ML Models

Abadi et al. [Ab16] introduced Differentially Private Stochastic Gradient Descent (DP-SGD), a modification of the SGD optimizer for DP. DP-SGD limits privacy loss by clipping gradients and adding Gaussian noise. The key parameter for private training is the privacy budget (ε), which sets the wanted DP privacy level and allows a translation to the needed clipping and noise parameters. In their work Ponomareva et al. [Po23] propose a guide for creating differentially-private machine learning applications and give an evaluation of privacy budget needs. They constitute that an $\varepsilon \leq 1$ provides strong formal privacy guarantees, while more realistic privacy guarantees more likely use an $\varepsilon \leq 10$, which they believe still provides a reasonable utility-privacy trade-off. They further define $\varepsilon > 10$ as giving just weak to no formal privacy guarantees. The authors argue these applications may still be protected against attacks, but without a real formal guarantee.

2.2 Likelihood Ratio Attack (LiRA)

We evaluate model vulnerability using the Likelihood Ratio Attack (LiRA) [Ca22], a Membership Inference Attack (MIA) [Sh17]. MIAs try to determine if a sample was part of the target model's training set by exploiting its confidence scores. The LiRA analyzes confidence distributions of shadow models, which involves training N shadow models on random samples, half including the query sample (x, y). These are IN and OUT models, respectively. Their confidence outputs are fitted to Gaussian distributions and the target model f is then queried on (x, y). A likelihood ratio test is performed comparing the target model's output to the IN/OUT distributions, producing a LiRA score indicating the membership probability.

3 Related Work

Prior work has investigated the influence of dataset characteristics on the vulnerability of ML models to MIAs. Shokri et al. [Sh17] found that models trained on datasets with more classes are more vulnerable, as the model must extract more discriminative features and thus retains more information about the training data. They also showed that classes with fewer samples are more susceptible to attack. Building on this, Truex et al. [Tr21] demonstrated that models trained on datasets with larger class sizes are more vulnerable to MIAs. They also found that datasets with higher in-class feature vector standard deviations lead to more accurate

attacks, as outlier samples have a greater influence on model training. Additionally, they showed that creating minority classes artificially increases the vulnerability of those classes in a previous work [Tr19]. Tonni et al. [To20] examined the effects of dataset size and class balance on MIA vulnerability in non-private settings. They found that smaller datasets are more susceptible to attack due to overfitting, and that minority classes are more vulnerable than majority classes. They also observed that higher data entropy decreases attack accuracy, as greater randomness makes it harder to infer membership information. Dealcala et al. [De24] recently looked into some other factors like batch size, dropout, and number of epochs. Overall, this prior work highlights the importance of dataset characteristics in determining the vulnerability of machine learning models to MIAs. However, previous work in this area has mostly focused on the characteristics that increase non-private model vulnerability, but has not evaluated the effects in private learning scenarios.

With their review, Ponomareva et al. [Po23] offer guidance on successfully training a private ML model. However, they are only focused on tweaking the actual ML training process for better results and miss out on including the underlying training data to create a complete and comprehensive picture of an ML application. Thus, while related work looked into general training and some data-related parameters, there is no comprehensive evaluation of a wider range of dataset characteristics, including both dataset-level (e.g., class size, class count, imbalance) and data-level (e.g., information density, color, class separability) properties. We fill this gap by providing a comparative analysis of the effects of image dataset characteristics across different privacy budgets ($\varepsilon = \infty$, $\varepsilon = 30$, $\varepsilon = 1$), providing insights into how their influence on model behavior plays changing roles with stronger privacy guarantees.

4 Experimental Setup

The experiments aim to determine how different dataset characteristics affect the utility and privacy of private ML models. These characteristics include dataset size, number of classes, class balance, and properties like information density of images, color and grayscale influence, and class similarity. The results provide guidance for practitioners and researchers working with private ML models. A schematic overview of the experiments' procedure is depicted in Fig. 2 and in this section, we give an overview on how we implemented this process. The experimental procedure has the following three steps, repeated with minor parameter changes to examine different aspects:

1. **Non-Private Model Creation**: Train non-private models on different datasets for image classification. The models share the same architecture but are trained on datasets differing in one characteristic (e.g., class size). This allows for comparing utility and privacy based on that characteristic.

2. **Private Model Training**: Convert non-private models to private models using a private learning algorithm to ensure DP. Train models with high and low privacy

Investigations:

- Utility-Privacy Tradeoff
- Influence of Dataset Structure
  - Class Size
  - Number of Classes
  - Dataset Imbalance
- Influence of Data
  - Information Density
  - Colored Images
  - Class Similarity

Datasets

Dataset Modifications

Data Measurements

Non-Private Model
$\varepsilon = \infty$

Private Model
$\varepsilon = 30$

Private Model
$\varepsilon = 1$

- Accuracy
- F1-Score
- Train-/ Test Accuracy Gap

Utility Measurement

Vulnerability Measurement

- MNIST
- FMNIST
- SVHN
- CIFAR-10
- CIFAR-100
- EMNIST

- Class Size
- Number of Classes
- Dataset Imbalance

- Entropy
- Compression Ratios
- Standard Deviation
- Fisher's Discriminant Ratio

- LiRA Attack
- AUC
- TPR@FPR=0.1
- TPR@FPR=0.001

Fig. 2: Illustration of aspects and procedures in this work's experiments.

budgets to assess utility loss as a trade-off for privacy. Compare prediction results to evaluate the dataset characteristics' influence on private model utility.

3. **Model Attacks**: Attack the private and non-private models using MIAs. Evaluate and compare the privacy of models with the same architecture but different training datasets. The attacks provide vulnerability metrics to understand how dataset characteristics affect private and non-private ML models' proneness to attacks.

4.1 Environment

All experiments run on a computing cluster using an NVIDIA Tesla V100 GPU. The software stack utilizes Python 3.9 with Google's ML library *TensorFlow* [Ma15] and the accompanying Keras framework for model training. In addition, the *TensorFlow Privacy*[4] library is used to create private models and also offers capabilities for attacking models with MIAs. To further emphasize the experiment's reproducibility, a fixed random seed value of 42 is used for all instances of random initialization, shuffling, sorting, or any other random methods executed on the datasets. Finally, reference code for all experiments is available from our repo at `https://github.com/luckyos-code/dataset-analysis-ppml`.

4.2 Datasets

This section introduces the datasets used in our experiments and analyses, highlighting their key characteristics. All the datasets are image datasets commonly used for multi-class

---

4 `https://github.com/tensorflow/privacy`

(a) MNIST dataset.   (b) FMNIST dataset.   (c) SVHN dataset.

(d) CIFAR-10 dataset.   (e) CIFAR-100 dataset.   (f) EMNIST dataset.
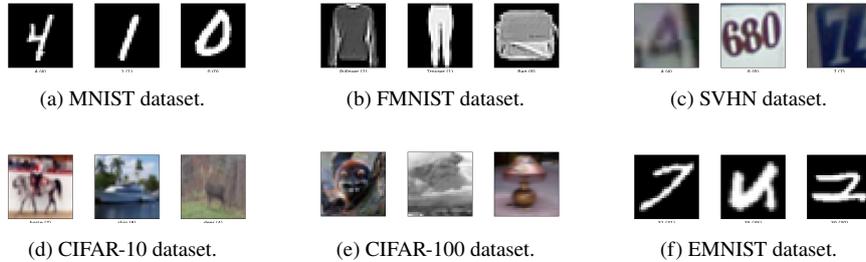
Fig. 3: Visual representation of random samples from the studied image datasets.

image classification tasks. While they do not directly contain sensitive information, they are among the most important benchmarking datasets in privacy-preserving machine learning (PPML) research, offering a diverse range of characteristics for our study [Ab16; Bo24; Ko16; Pa18]. In Fig. 3 we provide sample images from each dataset.

MNIST. The Modified National Institute of Standards and Technology (MNIST) dataset [Le98] is known for its simple structure and small image size and was originally created for a document recognition task. It contains 10 classes representing handwritten digits with labels from 0 to 9. The images as shown in Fig. 3a, are in grayscale and have a size of 28×28px. In total there are 70,000 images in the dataset, split up in 60,000 images for training and 10,000 images for testing. The MNIST dataset is often used as a benchmark for (private) image classification tasks [Ab16; La23; Un21] or to analyze the different learning approaches [KB14; PW17; Sr14]—making it a perfect basic candidate for our experiments.

FMNIST. The MNIST dataset is regularly criticized for its simplicity and age, since ML developed much since its initial release. Therefore, a more complex benchmarking dataset was needed, resulting in the Fashion MNIST (FMNIST) dataset [XRV17]. The dataset shares the same features as the MNIST dataset in all but the more complex images depicting products from an online fashion store and have a much wider variety of shapes and textures than the MNIST dataset, as is shown in Fig. 3b. Thanks to its matching structure, FMNIST is a suitable choice for a direct comparsion to the MNIST dataset. Since it contains a wider variety in patterns and textures it can be considered as a more complex task.

SVHN. The Street View House Numbers (SVHN) dataset [Ne11] contains cropped RGB images of house numbers from real Google Street View[5] images, with examples given in Fig. 3c. It has 10 classes, each representing a digit, and features about 73,000 training and 26,000 test images with 32×32px in size. The task is comparable to MNIST, but in contrast to the MNIST or FMNIST datasets, this dataset now contains RGB color images and the recognition task is again harder than the MNIST task [Ne11].

---

5 https://www.google.com/streetview/

CIFAR-10. With the Canadian Institute For Advanced Research (CIFAR)-10 [Kr09], we introduce another well-known RGB image classification dataset. It again consists of 10 classes and only a small image size of 32x32px, however, the depicted objects in these classes are real-life objects like horses, ships, deer, trucks, etc., shown in colorful images. Each class contains exactly 6,000 images and the resulting balanced dataset provides 50,000 samples for training and 10,000 for testing. Due to its color images, complex patterns, and non-uniform backgrounds, the multi-class image classification on CIFAR-10 is considered as an even more difficult task than with the FMNIST dataset [Ko18]. An excerpt of a few images from the dataset are shown in Fig. 3d. While featuring an increasingly complex task, the dataset again keeps a similar image size and class count as the MNIST and FMNIST datasets, which is an important aspect for explainable results.

CIFAR-100. The CIFAR-100 dataset [Kr09] is an extension to the CIFAR-10 dataset and while they both contain RGB color images with a size of 32×32px, as seen in Fig. 3e, the CIFAR-100 instead now consists of 100 classes with 600 images per class. This leads to an 10× increase in classes and an according decrease in samples per class. Keeping all factors similar to the CIFAR-10 and just drastically changing the amount of classes allows studying the influence of the number of classes on the model's performance and vulnerability.

EMNIST. The Extended MNIST (EMNIST) dataset [Co17] is an extension to the MNIST dataset that contains the complete NIST Special database 19 [GH95]. This is the database MNIST was derived from and means that the EMNIST contains exactly the same type of images. However, instead of only using handwritten digits, EMNIST additionally features uppercase and lowercase handwritten letters, with examples given in Fig. 3f. The dataset comes in different variations and we decided on the balanced version of the By_Merge variation [Co17], which results in 47 classes for 131,600 images. With similar image complexity as the MNIST dataset but providing more classes, EMNIST fits the same spot as the CIFAR-100 dataset enabling an isolated study regarding the class count.

4.3 Measuring Model Utility and Vulnerability

Measuring model utility helps assess the impact of dataset properties on both non-private and private learning. It is essential for balancing model privacy and utility. We use accuracy and F1-score to evaluate our multi-class classification tasks, with F1-score being preferred for evaluating unbalanced datasets [Hr05; SJS06; VGR20]. Model accuracy is also used to calculate the train-test accuracy gap, indicating overfitting [HRS15].

Practically measuring model privacy is crucial, as our theoretical DP-guarantees just provide an upper bound on possible information leakage and should thus not be taken as the single true indicator of privacy risk [Ra18]. MIAs can help evaluate this risk by determining a practical risk level through actual attacks, which can then be connected to our different privacy levels and datasets [Ma21]. We use the proposed offline version of the white-box

LiRA MIA [Ca22] with 32 shadow models. The offline attack eliminates the need for the IN shadow models described in Sect. 2.2.

Three metrics measure the attack's effectiveness: Receiver Operator Characteristics (ROC)-Area Under Curve (AUC), True Positive Rate (TPR) @ 0.1 False Positive Rate (FPR), and TPR@0.001 FPR, which were suggested by Carlini et al. [Ca22]. They promote measuring the TPR at a low fixed FPR, since a low FPR rate is most important for an attacker. In general, a ROC curve visualizes the relationship between an attack's TPR and FPR, which is then captured across all FPR values using the AUC. All three metrics have baseline values constituting a randomly guessing attacker, which would translate to perfect privacy. The baseline for the AUC metric is 0.5, while the baselines for TPR are 0.1 and 0.001, respectively. Finally, each shadow model gives a single attack result for its part of the dataset and we therefore give average results over all shadow models.

4.4 Experiments

In this part, we detail our evaluation strategy for relevant dataset characteristics. A general overview of the experiments is given in Fig. 2. The experiment settings are intended to provide information to help researchers and practitioners working with PPML. The results should help them in designing a secure private ML model faster by considering the influence of different aspects of a dataset in the process. The experiments explore which dataset characteristics increase or decrease the vulnerability and utility of private and non-private ML models, which is evaluated by the metrics described in Sect. 4.3. For the private model experiments, we further distinguish between two privacy scenarios using different privacy budgets ($\varepsilon = \{1, 30\}$).

The dataset characteristics are divided into two different levels depending on what kind of modifcations are applied. The first set of characteristics considers the *dataset-level*, which focuses on the overall dataset structure with statistics like number of samples, class count and class imbalance. The second level consists of *data-level* modifications, which are characteristics that e.g., measure data complexity, the influence of color information, and class separability. While we can generally change dataset-level factors by using fitting slices of the dataset, modifying most underlying data-level characteristics like data complexity is much more difficult. To avoid this problem, we instead use a selection of differing datasets as presented in Sect. 4.2 to better evaluate these data-level characteristics.

For each setting, we define a set of modifications with changing parameters that we apply only to a dataset's training split, leaving the test split untouched for our evaluating metrics. For each modified dataset version, we train a non-private and two private models with our different privacy budgets $\varepsilon = 1$ and $\varepsilon = 30$. On these models we then measure the impact of the varying dataset setups for each modification and privacy level, by first calculating their utility scores and then by attacking them with the LiRA attack. In the following, we provide information on our used modifications.

4.4.1 Dataset-Level Investigations

The dataset-level investigations are a series of experiments to assess how overall dataset statistics influence the private learning of ML models.

Class size. The first dataset modification introduces a reduction and normalization of the class size, i.e., the number of samples per class. It reduces the number of samples in each class to a fixed value, which in our experiments is called c. This reduction is done by randomly removing samples per class until the specified number of samples per class is reached. This also means, that we achieve a perfect class balance, since all classes are equal in their sample counts. The datasets used in this experiment are MNIST, FMNIST, SVHN and CIFAR-10, for which we run several experiments with different class sizes. In total, there are eight class sizes, which successively reduce the size to 5000, 4000, 3000, 2000, 1000, 500, 100 and 50 samples per class. The biggest size of $c = 5000$ is the largest common class size across all datasets, acting as the baseline experiment. This amount is enough to see if bigger classes reduce the privacy risk due to more available data.

Class count. The class count modification changes the number of classes in a dataset by purposefully reducing them, which is done by deleting all samples of a set number of existing classes. The resulting class count is denoted in the parameter n and to make the deletion process deterministic for all datasets, first, all available class labels are retrieved and placed in an alphanumerically ordered list. From this ordered list, only the first n labels are kept and the rest are discarded together with their respective samples in the dataset.

This experiment is divided into two sub-parts depending on the original dataset classes available. The first part, we use the datasets with fewer classes, namely the MNIST, FMNIST, SVHN and CIFAR-10. Their class counts are each reduced from 10 down to 3, which is the border for still keeping a multi-class classification task. The second part aims at the CIFAR-100 and EMNIST, which consist of many more classes. The baseline is based on EMNIST, which contains 47 classes, while CIFAR-100 has even more at 100 classes. With this, we reduce the classes over a larger range than before to generate further insights. We start at 47 classes, which is the baseline as it is the maximum for EMNIST, and reduce the class size in steps of five until it reaches three classes.

Class imbalance. Class imbalance describes the issue of a skewed data distribution, meaning that some classes contain more samples than others, which results in minority and majority classes. While minority classes are classes that are underrepresented, by having much lesser samples, majority classes are overrepresented and have much more samples than the minority classes. In this operation, we remove data samples to create an artificial class imbalance. For this we introduce the imbalance factor i, which we keep in a $[0, 1]$ range. Here, $i = 0$ means that there is no class imbalance, and $i = 1$ translates to the highest possible class imbalance.

Furthermore, this modification works in two modes. Both modes use the same datasets,

| (a) $i = 0.3$, linear. | (b) $i = 0.9$, linear. | (c) $i = 0.3$, normal. | (d) $i = 0.9$, normal. |
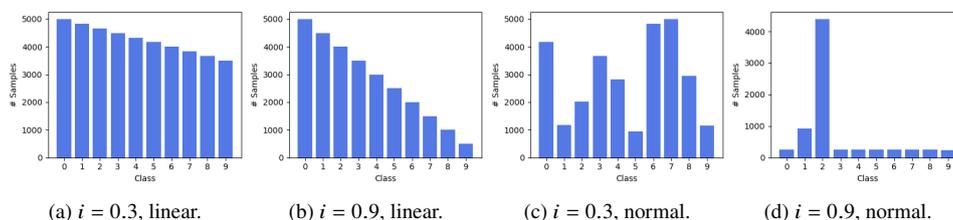
Fig. 4: Visualization of the dataset class distribution after applying the dataset imbalance modification in linear and normal mode with varying imbalance factors $i = 0.3$ and $i = 0.9$.

which are MNIST, FMNIST, SVHN and CIFAR-10, with five values for i at 0.0 (baseline), 0.1, 0.3, 0.6, and 0.9. Before applying the imbalance modification, we reduce all the class sizes to 5000 to start with perfectly balanced and equal-sized datasets.

The first mode applies imbalance in *linear* mode, which means that the classes of the dataset decrease linearly in size, and no two classes have the same size. This is done by first sorting all classes by their current class size. Then the smallest class size is multiplied by $1 - i$, which is the final new class size for the smallest class in the dataset. Now the class size values between the first and largest class, and the last and smallest class are set linearly ascending with equal intervals. The resulting distribution of samples in linear mode for $i = \{0.3, 0.9\}$ is visualized in Fig. 4a and 4b.

The second is the *normal* mode, which creates imbalance by randomly removing samples based on a normal distribution, with its mean as mean $= 1 - i$ and the standard deviation as std $= i$. We further clip the drawn values at 1.0 and 0.05. The sampled values are multiplied by the current class size, resulting in a new and possibly reduced size. The modified distribution in normal mode is visualized in Fig. 4c and 4d. The visualization shows the randomness of normal mode imbalance compared to linear mode.

4.4.2 Data-Level Investigations

While the dataset-level experiments (see Sect. 4.4.1) investigate the model behavior in relation to the structural properties of the dataset, the data-level experiments observe how the data itself influences the private and non-private ML models. The focus of the data-level investigations is to find and compute quantifiable metrics that describe the image data itself.

Information density. One way to describe the complexity of datasets is to measure the density of information in the data. Comparing the MNIST and CIFAR-10 images gives a general sense of what is meant by this characteristic. The MNIST dataset consists of grayscale pixels forming simple structures, whereas the CIFAR-10 dataset has a much wider variety of pixel brightness, complex textures, and patterns. In short, the CIFAR-10 images appear more complex than the MNIST images. We quantify image complexity by

using two methods. The first method is to calculate the Shannon entropy [Sh48], which defines as a measure of randomness inside the data. A high entropy value indicates that there is more disorder or randomness in the data, while a lower entropy value means that the data is more ordered and predictable. The idea is that more information in a dataset translates to less uncertainty. We calculate the entropy H image-wise as follows: $H(X) = -\sum_{x \in X} p(x) \log p(x)$. Where X is the set of all pixel values in an image and $p(x)$ describes the probability that a pixel with value x occurs. This equation works very well for grayscale images, however, calculating the entropy (H) of color images requires an extra step, in which we calculate an averaged entropy over the three color channels.

The second measure of information density is the compression ratio. In [YW13], the authors argue that Shannon's entropy is not a good measure of image complexity because it does not account for spatial structure. They therefore define a compression ratio measure, which is defined as: $CR = \frac{s(I)}{s(C(I))}$. Where $s(I)$ is the uncompressed image size in bytes and $s(C(I))$ is the compressed image size in bytes. This ratio is an indicator of how much the image I can be compressed. The idea behind this measure is that noise patterns in images that do not contain much information, but consist of random patterns, cannot be compressed as efficiently as patterns that contain actual visual information. We use two different compression methods, a lossy and a lossless compression. A lossy compression method loses some image information during image compression, while a lossless compression method can compress images without losing any information. The lossy method used is the Joint Photographic Experts Group (JPEG) [OB05], which utilizes visual features by extracting frequency information from the image. The second approach uses the lossless Portable Network Graphics (PNG) image compression [Bo97], which compared to the JPEG, works more on the pixel level of the image rather than the visual characteristics.

Color. For experimenting with color, we take SVHN and CIFAR-10, once with their original color and once in a modified grayscale version. Three information channel (RGB) should provide more information than one channel of information (grayscale). To transform color images to grayscale, we multiply each RGB color channel (red, green, blue) by a specific weight representing the wavelength of the color. These values are then summed up to represent the grayscale value for that pixel.

Class similarity. Class similarity describes the similarity of data samples within a class or between classes. The intuition behind analyzing class similarity is to roughly estimate task difficulty of multi-class classification problems. If the samples between classes are similar to each other, the classification task may become harder, since the rules needed to classify the data samples may be harder to learn for an ML model. On the other hand, if many samples within a class are very similar, they are easier to classify as belonging to that class because they share more common attributes. Class similarity is analyzed using two different methods. The first method follows the work of Truex et al. [Tr21] and calculates the in-class feature vector Standard Deviation (STD) of the dataset. The motivation behind using the STD is to calculate how much the samples within a group differ from each other on average, which is an indicator of how similar the data is. The second method of class similarity calculates the

Fisher Discriminant Ratio (FDR). This ratio measures the degree to which the classes of a dataset are separable. The FDR is known from its use in Fisher's Linear Discriminant Analysis (LDA) as a measure to be maximized by the LDA procedure [Fi36]. Basically, the FDR is the ratio of the between-class variance to the within-class variance [LW14]. The motivation for analyzing class separability is that classes that are more similar to each other are harder to separate. Thus, a lower FDR value for a dataset could indicate that this dataset is harder to classify than one with a higher FDR value.

4.5 Pre-processing

Regarding pre-processing, we detail the needed modifications and augmentations applied to the training data to fit the model's input shape and improve utility. First, all datasets are split into train and test sets, using their provided standard splits to ensure representativeness and comparability with other work. With such differing inputs from our multiple datasets, we have to consolidate them into some common input space. Thus, all images are scaled before training to match the required 32×32 input shape for our model described in Sect. 4.6. Datasets with smaller dimensions, such as MNIST (28×28), are upscaled accordingly. Next, 8-bit color image values are normalized to the [0, 1] range to improve convergence, particularly with Rectified Linear Unit (ReLU) activation functions. For grayscale datasets like MNIST or Fashion-MNIST, which have one channel, two additional identical channels are added to create a 32×32×3 input shape to match the model's requirements, which has to classify images from all datasets. Data augmentation is a common tool for enhancing a model's generalization ability by combating overfitting. We apply horizontal flipping based on Carlini et al. [Ca22], where we randomly flip 50% of images on the vertical axis.

4.6 Model Architecture and Training

Our chosen architecture is a CNN similar to LeNet-5 [Le98] that we modified to better suit our tasks. The model starts with a *RandomFlip* layer, which randomly applies horizontal flips on the input to ensure the data augmentation from Sect. 4.5. The rest of the model comprises three convolutional, one hidden dense, and one output dense layer, which are each connected by group normalization and max pooling layers. We decided on a basic but powerful architecture to allow maximum generalizability to CNNs. Choosing differing or too complex models can distort the findings due to these newly introduced factors. Following the guidelines from Ponomareva et al. [Po23], we performed a hyperparameter search. This results in using the Adam optimizer with a learning rate of $\alpha = 0.005$ and batch size of 256 over 30 epochs. For private training, we shift to DP-Adam, a differentially-private alternative, with microbatches of 256 and a clipping norm of 1.0. The noise parameters are dynamically determined according to our privacy budgets $\varepsilon = 1$ and $\varepsilon = 30$, since they also depend on the training data. For example, with 60,000 MNIST samples, batch size 256, and 30 epochs, the noise multipliers are $\sigma = 0.431$ for $\varepsilon = 30$ and $\sigma = 1.626$ for $\varepsilon = 1$.

(a) Non-private ($\varepsilon = \infty$) model. (b) Private ($\varepsilon = 30$) model. (c) Private ($\varepsilon = 1$) model.
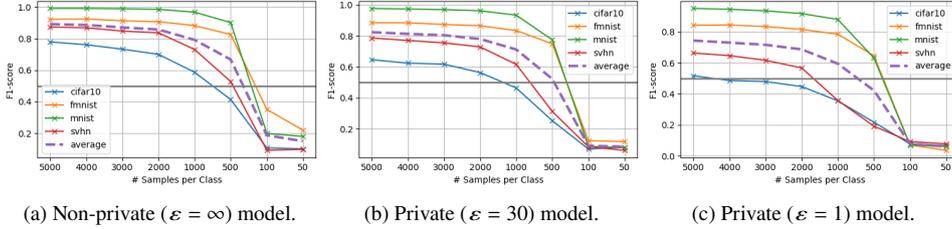
Fig. 5: F1-scores for non-private and private models on different datasets with modified class sizes.

(a) Non-private ($\varepsilon = \infty$) model. (b) Private ($\varepsilon = 30$) model. (c) Private ($\varepsilon = 1$) model.
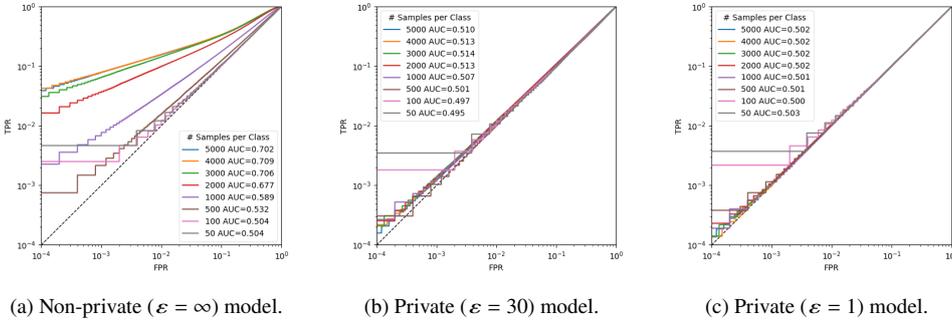
Fig. 6: Attack ROC curves for averaged non-private and private models with modified class sizes.

5 Evaluation

This section presents the experimental results from Sect. 4, focusing on the dataset characteristics that influence the behavior of the ML model, assessed through utility (accuracy, F1-score) and vulnerability (LiRA MIA). We examine models at three different privacy levels: non-private models at $\varepsilon = \infty$, private models with $\varepsilon = 30$, and $\varepsilon = 1$. We use multiple datasets, varying in the difficulty of their multi-class image classification tasks but having similar structures (see Sect. 4.2).

5.1 Dataset-Level Results

This part investigates how dataset characteristics, such as class size, class count, and dataset imbalance, affect model utility and vulnerability.

Class size. For evaluating the utility over our proposed class size reductions, we use Fig. 5, which shows the achieved F1-scores for each class size and privacy budget regarding different datasets. We generally observe that model utility decreases with fewer samples per class. For non-private models, the average F1-score dropped from 0.89 (5000 samples/class) to 0.19 (100 samples/class) and we find similar behavior for the private models. However, they additionally battle with reduced overall utility due to their utility-privacy trade-off.
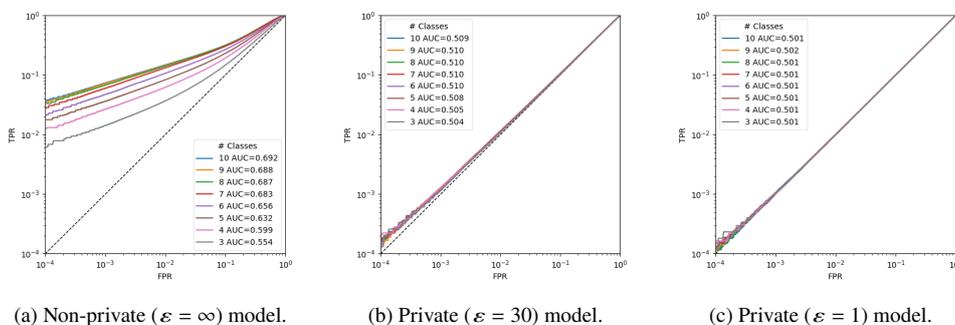
(a) Non-private ($\varepsilon = \infty$) model. (b) Private ($\varepsilon = 30$) model. (c) Private ($\varepsilon = 1$) model.

Fig. 7: Attack ROC curves for averaged non-private and private models with modified class counts.

Our vulnerability test in Fig. 6 presents attack susceptibility across datasets using ROC curves. At 5000 samples per class, CIFAR-10 is generally most vulnerable and MNIST is least. In our non-private models, we can observe an overall trend of reduced vulnerability in relation to reduced class sizes. From Fig. 5 we can see that the average utility still holds up until 2000 samples before steadily declining from 1000 samples onwards. Here, the vulnerability advantages at 2000 samples could signify a good utility-privacy trade-off, but only from 1000 samples and lower, we can notice a major threat reduction by about half. With privacy ($\varepsilon = 30$ and $\varepsilon = 1$), vulnerability significantly decreases, making attacks less effective. Even our weaker privacy at $\varepsilon = 30$ is able to effectively limit attack threats, while $\varepsilon = 1$ models even demonstrate near-random attack effectiveness. Nonetheless, we find two outliers for class sizes of 50 and 100 that still show vulnerability at low TPR even in our strictly private model, which is due to the model's very low performance in these cases. The model's results are just random guessing and are therefore not really meaningful.

Class count. In this experiment, performance intuitively increases when lowering the class count, since the classification task gets easier. Therefore, we find the non-private models' average F1-score continuously increasing from 0.90 (10 classes) to 0.97 (3 classes) and 0.68 (47 classes) to 0.81 (3 classes). Further, the more complex tasks like CIFAR-10 and SVHN see greater benefits from reducing classes. The privacy results in Fig. 7 show that non-private models become less vulnerable as class count decreases. We just focus on reducing from 10 classes, since the 47 class case gives the same patterns. Private models generally show reduced vulnerability, with $\varepsilon = 1$ models exhibiting negligible vulnerability changes because all models are very close to the optimum even at very low TPR. In summary, reducing class count increases utility and reduces vulnerability, with private training significantly enhancing security across all scenarios.

Class imbalance. In terms of utility, increasing imbalance in linear mode linearly decreases overall F1-scores by 4% due to the underperformance of the created minority classes, which is amplified in the private models with 8% and 14% at $\varepsilon = 30$ and $\varepsilon = 1$, respectively. Normal mode showed especially devastating utility loss (29%, 39%, 48%) because the
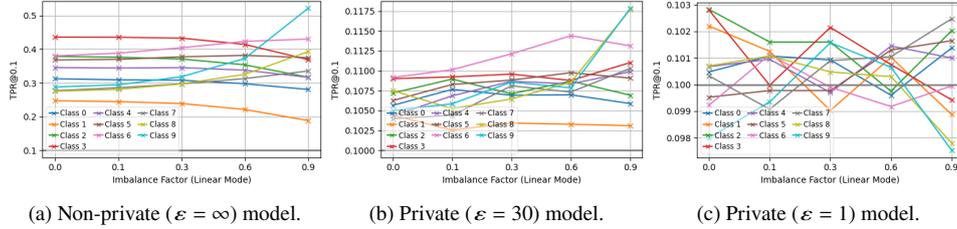
(a) Non-private ($\varepsilon = \infty$) model. (b) Private ($\varepsilon = 30$) model. (c) Private ($\varepsilon = 1$) model.

Fig. 8: Average class-wise attack TPR@0.1 results for datasets with varying class imbalance (linear mode) and privacy budgets. Note the scaling of the y-axes between different privacy budgets.
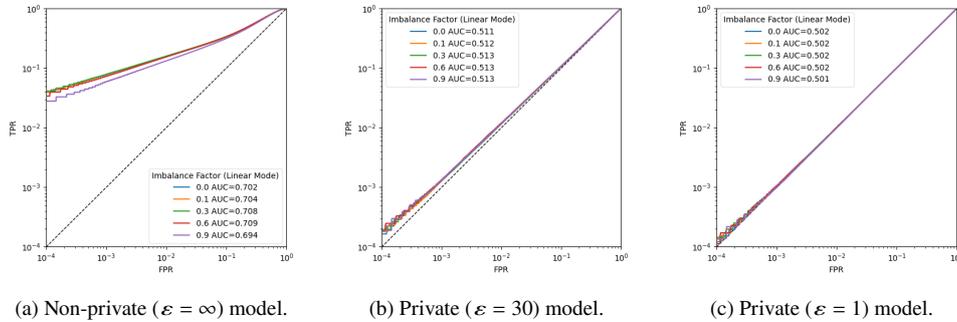
(a) Non-private ($\varepsilon = \infty$) model. (b) Private ($\varepsilon = 30$) model. (c) Private ($\varepsilon = 1$) model.

Fig. 9: Attack ROC curves for averaged models with modified class imbalance (linear mode).

normal distribution randomness leads to having mostly minority classes and keeping just one or two bigger classes at higher imbalance factors, which shows to be very challenging. We can thus only conclude that such data setups are generally infeasible for equally weighted classification tasks, since usable performance across all classes is essential.

Minority classes created further problems when looking at the linear mode attack TPR@0.1 for each class in Fig. 8, where we can clearly see the minority classes spiking in TPR for the non-private and $\varepsilon = 30$ models, increasing their threat level compared to the other classes. Important to note however, that the maximum for the private model is significantly lower and therefore a less pronounced increase. For $\varepsilon = 1$, the strict DP successfully obfuscates the produced minority classes, resulting in very low TPR and no recognizable outlier. In Fig. 9 we can compare these results to the average ROC curves over all classes, which paints a different picture of vulnerability. When focusing on averages, we no longer see significant differences in threat levels and instead only notice slight changes in relation to shifting imbalance. We instead only record the notable changes due to DP in our privacy models. An increasing class imbalance thus reduces both utility and privacy due to minority classes, which is however not always clearly visible when just focusing on averages.

| Dataset | Entropy | Compression | | $\varepsilon = \infty$ | | $\varepsilon = 30$ | | $\varepsilon = 1$ | |
		JPEG	PNG	F1	AUC	F1	AUC	F1	AUC
MNIST	0.20	0.52	0.35	0.99	0.54	0.98	0.50	0.95	0.50
FMNIST	0.51	0.57	0.65	0.93	0.64	0.89	0.51	0.85	0.50
SVHN	0.82	0.16	0.56	0.89	0.71	0.79	0.51	0.67	0.50
—ıı—(gray)	0.79	0.33	0.64	0.88	0.71	0.79	0.51	0.66	0.50
CIFAR-10	0.89	0.19	0.73	0.78	0.86	0.64	0.52	0.51	0.50
—ıı—(gray)	0.86	0.43	0.81	0.76	0.87	0.62	0.51	0.48	0.50

Tab. 1: This overview presents the information density measurements together with each dataset's utility (F1-score) and vulnerability (AUC) across privacy budgets. A privacy budget of $\varepsilon = \infty$ indicates a non-private model and an $AUC = 0.50$ translates to no vulnerability.

5.2 Data-Level Results

We now analyse the influence of data-level properties such as information density, color, and class separability on model behavior. Apart from information density, Tab. 1 also presents our datasets' overall results regarding model utility (F1-score) and vulnerability (AUC) across the three privacy levels. Regarding these DP levels, we can see the general expected trend of reduced utility with stricter budgets. This however also results in really strong defense against our attacks, where models with $\varepsilon = 30$ already reduce their risks to only minor deviations from the perfect score of 0.50 AUC. Models at $\varepsilon = 1$ are private enough to even guarantee perfect scores for all our datasets. Comparing between datasets, our models trained on MNIST exhibit the highest non-private utility at 0.99, with a clear downwards trend as we increase the complexity of the classification task until we reach 0.78 on CIFAR-10. Shifting to the private models with $\varepsilon = 30$ and $\varepsilon = 1$, we see just a slight drop to a low of 0.95 in MNIST, while the more complex tasks see a steeper decrease in utility, like CIFAR-10 falling to 0.51 F1. Transitioning to vulnerability we see the same trend for non-private models, where MNIST is least vulnerable with an AUC of 0.54 and AUC increases up to the CIFAR-10 being most vulnerable at 0.87 AUC. However, as stated before, private learning significantly reduces vulnerability across all datasets and already almost fully removes these discrepancies at the weaker level of $\varepsilon = 30$. It will be interesting to see if we can confirm these observations using our information density, color, and class separability metrics.

Information density. We again use the results from Tab. 1 for evaluating our information density metrics, where we gathered each datasets entropy and JPEG/PNG compression rates. These values are accompanied by model utility (F1-score) and vulnerability (AUC) results across our three privacy levels. Higher shannon entropy seems to correlate with increased vulnerability. MNIST had the lowest entropy and vulnerability, while the other higher entropy datasets each increased in vulnerability up to the CIFAR-10, which showed the highest risk. The JPEG and PNG compression ratios do not show the same clear

correlations with vulnerability or utility. Lower JPEG compression indicates a relation to higher vulnerability in SVHN and CIFAR-10, which is however undermined by their gray image variants that show increased compression ratio at the same AUC results.

Color. When investigating the influence of color using our gray datasets in Tab. 1, we find only slight differences in utility between the model on grayscale or color data. The same holds for vulnerability, where we see just a minimal difference when using grayscale, making practically no impact on our model results. Therefore, color does not seem to hold significant influence or potential when optimizing our data for private training.

Class separability. For analyzing class separability we still use the utility and vulnerability results from Tab. 1 but now try to link them to the FDR and STD results given in Fig. 10. We clearly notice that MNIST has the highest FDR, indicating better separability between its classes. We find both, MNIST and FMNIST, showing higher FDR, while in turn exhibiting lower vulnerability and utility loss than the other datasets. On the lower end of FDR, the correlation seems to vanish, since SVHN

(a) FDR

(b) Average STD

Fig. 10: Visualization of class separability measurement results.

and CIFAR-10 show to be close in FDR, while their AUC is clearly set apart by a 0.15 difference. In the same vein, the gray variants do not match the others regarding their FDR and AUC rankings. Regarding STD, we find the same results, where again the MNIST and FMNIST with low vulnerability can be successfully separated from the other datasets due to their high STD values. Among the other datasets we again do see the same trend, where vulnerability does not follow their STD differences.

5.3 Discussion

We first want to summarize our overall findings before giving our extracted practical guidelines. Starting with the class size experiment in Tab. 2, reducing the samples per class decreases utility for both private and non-private models, with private models being more sensitive. Overfitting is observed in non-private models and slightly in private models with $\varepsilon = 30$. The most private model ($\varepsilon = 1$) showed no overfitting. Vulnerability initially increases with fewer samples and then decreases, linked to the overfitting effect. Reducing the number of classes increases model utility across all models, with a more pronounced effect in models with lower privacy budgets. Higher class counts tend to make models more vulnerable, whereas fewer classes reduce this vulnerability, especially in non-private models. Although it might seem counterintuitive—since more classes imply more data to obscure sensitive information—it is in line with Shokri et al. [Sh17] and the key insight, that models are generally less confident when classifying a larger number of classes. This

Tab. 2: Overview of dataset-level investigation results. An $\varepsilon = \infty$ represents the non-private models.

Experiment	$\varepsilon = \infty$	$\varepsilon = 30$	$\varepsilon = 1$
Class Size – Decrease Number of Samples per Classs	• decreasing utility, overfitting effect appears • vulnerability increase at overfitting, else decreasing (due to low utility)	• utility starts decreasing earlier, smaller overfitting effect • small vulnerability increase at overfitting, else decreasing	• utility starts decreasing even earlier, no overfitting effect • no changes in vulnerability
Class Count – Decrease Number of Classes	• utility increase, decreasing overfitting • strong vulnerability decrease	• more increasing utility, no decreasing overfitting • small vulnerability decrease for some datasets	• most increase in utility • no changes in vulnerability
Class Imbalance – Increase Dataset Imbalance	• decreasing utility (mostly minority classes) • vulnerability increase of minority classes	• stronger utility decrease of minority classes • smaller vulnerability increase of minority classes	• strongest utility decrease of minority classes • no changes in vulnerability

Tab. 3: Overview of the data-level investigation findings.

Investigated Aspect	Observed Effects
Entropy	higher values → more vulnerable & less utility in private learning
JPEG Compression Ratio	higher values → less complex images & less vulnerable
PNG Compression Ratio	no recognizable effect
FDR	higher values → less utility loss in private learning
STD	lower values → higher vulnerability
Removal of Color	slightly worse utility

reduced confidence makes it easier to distinguish between seen and unseen samples, where the confidence usually spikes in previously seen ones. Conversely, with fewer classes, models exhibit higher overall confidence due to the simpler classification task, reducing the confidence gap between seen and unseen, and thus lowering the attack's success. Finally, for class imbalance, we notice that model utility decreases with increasing imbalance, especially for private models. Minority classes are more vulnerable, with this effect being smaller in private models with $\varepsilon = 30$ and not observed in models with $\varepsilon = 1$.

For the data-level results presented in Tab. 3, we first look at the entropy. The datasets with higher entropy are more vulnerable and lose more utility with private learning. However, we can only find these results in the average entropy over the entire datasets, whereas a

		Compression		Separability		$\varepsilon = \infty$		$\varepsilon = 30$		$\varepsilon = 1$	
Size	Entr.	JPEG	PNG	FDR	STD	F1	AUC	F1	AUC	F1	AUC
7.2k	0.94	0.03	0.19	<0.01	0.25	0.92	0.64	0.85	0.53	0.79	0.51

Tab. 4: Results for the practical privacy scenario on the more sensitive COVID-19 data. We look at some characteristics, utility (F1-score), and vulnerability (AUC) across privacy budgets. A privacy budget of $\varepsilon = \infty$ indicates a non-private model and an $AUC = 0.50$ translates to no vulnerability.

class-wise comparison between the entropy and attack proneness of individual classes does not show any correlation. The JPEG compression ratio can instead help to estimate the complexity of an image dataset and by that also its task difficulty. This is supported with higher ratios indicating higher utility and lower vulnerability in our models. In terms of class separability, we find that the FDR can provide a general indication of model vulnerability. Datasets with high FDR are less vulnerable and lose less utility with private learning. The average dataset STD is also related to vulnerability, with lower STD datasets being more likely to be vulnerable. Converting colored datasets to grayscale slightly reduces utility and marginally influences vulnerability. This can be linked to the decreased entropy, altered JPEG ratio, and FDR values, though these metrics do not consistently predict the vulnerability changes. Especially the color influence experiments show that the data-level metrics and their effect on vulnerability and utility can only be used as general indicators.

5.3.1 Implications

No single data-related characteristic fully describes how a dataset affects a private model's performance or vulnerability. Instead, these metrics must be combined to estimate the model's behavior when trained on a specific dataset. We untangled five key rules of thumb regarding dataset characteristics for building private machine learning applications: (1) Use DP-private learning: In sensitive contexts, apply DP-private learning whenever possible. We find that even with a larger privacy budget, attack success is significantly reduced, balancing vulnerability differences between models trained on different datasets. (2) Amount of data: Fewer samples per class also reduce the overall data, leading to overfitting and increased vulnerability. Prioritize acquiring data where possible, especially for smaller classes, over optimizing the training process. (3) Number of classes: More classes increase the proneness to MIAs. Reduce the number of classes to a feasible minimum for the task or use DP-private learning to mitigate this vulnerability. (4) Dataset imbalance: Imbalance affects individual class vulnerability, especially for minority classes. Balance the dataset manually or use private models to equalize vulnerability across classes. (5) Image complexity: More complex datasets have a worse utility-privacy trade-off and higher vulnerability. Analyzing entropy, FDR, STD, and JPEG compression ratio can help estimate dataset complexity. These metrics are useful for comparing datasets and assessing how models change with new data.

5.3.2 Comparison to Practical Privacy Scenario

Before concluding, we compare our findings to a practical test scenario. A limitation of our analysis is that it was conducted solely on benchmarking datasets, which, while fitting for our study, lack obvious privacy implications (Sect. 4.2). To address this, we test our results in a privacy-conscious setting by drawing on previous work [La23], where we investigated private COVID-19 detection. Unlike our benchmarking datasets, this binary task involved sensitive medical images, where a successful MIA could reveal a person's COVID-19 status.

Considering our rules (1)–(5) from Sect. 5.3.1 and the characteristics of the COVID-19 dataset [Ch20; Ra21]—(2) limited training data, (3) two classes, (4) medium imbalance (50% more normal than COVID-19 cases), and (5) high complexity with low compression and FDR (see Tab. 4)—we expect a challenging utility-privacy trade-off and vulnerability to MIAs, though the low class count (3) might provide some relief. The results in Tab. 4 confirm medium vulnerability at $\varepsilon = \infty$ and a noticeable utility drop with stricter DP budgets on the COVID-19 model. Despite more challenging data-level statistics compared to the FMNIST dataset (see Tab. 1), both show roughly equal results, likely due to the 2- vs. 10-class task advantage. Using our suggestions, although we cannot collect more data (2)+(4) or further reduce classes (3), we can apply strategy (1) and successfully use a lower DP level of $\varepsilon = 30$ at a low $AUC = 0.53$, maintaining both utility and defense capabilities.

6 Conclusion

In this work, we take into account various dataset characteristics to provide guidance for implementing differentially-private image classification models. The primary goal is to help researchers and practitioners in determining a priori, if DP is worthwhile and what needs to be considered for specific datasets. In our experiments, we assess ML model utility and vulnerability to MIAs across different datasets, while relying on varying privacy budgets for DP ($\varepsilon = \{\infty, 30, 1\}$). Our derived implications for effectively using our results in engineering private ML models are summarized in Sect. 5.3.1 and our findings have been applied to a practical scenario in Sect. 5.3.2. We want to give data-related optimization a bigger stage in PPML, where we might not be able to directly access private data but instead have to rely on general data metrics. A key aspect of our analysis is that our models effectively mitigate most of the MIA threat across all datasets with a modest privacy budget of $\varepsilon = 30$, achieving a more practical utility-privacy trade-off at low risk. By considering the influence of different aspects of a dataset in addition to the actual training process, the broader picture allows steering in the most effective directions. Future work could explore the influence of model architectures on MIA threat and extend this study to non-image data.

References

[Ab16]   Abadi, M.; Chu, A.; Goodfellow, I.; McMahan, H. B.; Mironov, I.; Talwar, K.; Zhang, L.: Deep Learning with Differential Privacy. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. ACM, Vienna Austria, pp. 308–318, 2016, ISBN: 978-1-4503-4139-4, DOI: 10.1145/2976749.2978318, URL: https://dl.acm.org/doi/10.1145/2976749.2978318.

[AC19]   Al-Rubaie, M.; Chang, J. M.: Privacy-Preserving Machine Learning: Threats and Solutions. IEEE Security & Privacy 17 (2), pp. 49–58, 2019, ISSN: 1540-7993, 1558-4046, DOI: 10.1109/MSEC.2018.2888775, URL: https://ieeexplore.ieee.org/document/8677282/.

[Bo24]   Boenisch, F.; Mühl, C.; Dziedzic, A.; Rinberg, R.; Papernot, N.: Have it your way: Individualized Privacy Assignment for DP-SGD. Advances in Neural Information Processing Systems 36, 2024.

[Bo97]   Boutell, T.: PNG (Portable Network Graphics) Specification Version 1.0. 1997.

[Ca22]   Carlini, N.; Chien, S.; Nasr, M.; Song, S.; Terzis, A.; Tramer, F.: Membership Inference Attacks From First Principles. In: 2022 IEEE Symposium on Security and Privacy (SP). IEEE, San Francisco, CA, USA, pp. 1897–1914, 2022, ISBN: 978-1-66541-316-9, DOI: 10.1109/SP46214.2022.9833649, URL: https://ieeexplore.ieee.org/document/9833649/.

[Ch20]   Chowdhury, M. E.; Rahman, T.; Khandakar, A.; Mazhar, R.; Kadir, M. A.; Mahbub, Z. B.; Islam, K. R.; Khan, M. S.; Iqbal, A.; Al Emadi, N., et al.: Can AI help in screening viral and COVID-19 pneumonia? Ieee Access 8, pp. 132665–132676, 2020.

[Co17]   Cohen, G.; Afshar, S.; Tapson, J.; van Schaik, A.: EMNIST: an extension of MNIST to handwritten letters. 2017, DOI: 10.48550/ARXIV.1702.05373, URL: https://arxiv.org/abs/1702.05373.

[De24]   Dealcala, D.; Mancera, G.; Morales, A.; Fierrez, J.; Tolosana, R.; Ortega-Garcia, J.: A Comprehensive Analysis of Factors Impacting Membership Inference. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Pp. 3585–3593, 2024.

[EC16]   European Parliament; Council of the European Union: 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 (General Data Protection Regulation), OJ L 119, 4.5.2016, p. 1–88, 2016, URL: https://data.europa.eu/eli/reg/2016/679/oj, visited on: 04/13/2023.

[Eu22]   European Data Protection Board: Hellenic DPA: Fines imposed to telecommunications companies due to personal data breach and illegal data processing, 2022, URL: https://edpb.europa.eu/news/national-news/2022/hellenic-dpa-fines-imposed-telecommunications-companies, visited on: 06/14/2023.

[Fi36]   Fisher, R. A.: THE USE OF MULTIPLE MEASUREMENTS IN TAXONOMIC PROBLEMS. en, Annals of Eugenics 7 (2), pp. 179–188, 1936, ISSN: 2050-1420, 2050-1439, DOI: 10.1111/j.1469-1809.1936.tb02137.x.

[Ge20]   Geiping, J.; Bauermeister, H.; Dröge, H.; Moeller, M.: Inverting Gradients – How easy is it to break privacy in federated learning? 2020, DOI: 10.48550/ARXIV.2003.14053, URL: https://arxiv.org/abs/2003.14053.

[GH95]   Grother, P. J.; Hanaoka, K.: NIST special database 19. Handprinted forms and characters database, National Institute of Standards and Technology 10, p. 69, 1995.

[Hr05]   Hripcsak, G.: Agreement, the F-Measure, and Reliability in Information Retrieval. en, Journal of the American Medical Informatics Association 12 (3), pp. 296–298, 2005, ISSN: 1067-5027, 1527-974X, DOI: 10.1197/jamia.M1733.

[HRS15]   Hardt, M.; Recht, B.; Singer, Y.: Train faster, generalize better: Stability of stochastic gradient descent. 2015, DOI: 10.48550/ARXIV.1509.01240, URL: https://arxiv.org/abs/1509.01240.

[KB14]   Kingma, D. P.; Ba, J.: Adam: A Method for Stochastic Optimization. 2014, DOI: 10.48550/ARXIV.1412.6980, URL: https://arxiv.org/abs/1412.6980.

[Ko16]   Konečný, J.; McMahan, H. B.; Yu, F. X.; Richtárik, P.; Suresh, A. T.; Bacon, D.: Federated learning: Strategies for improving communication efficiency. arXiv preprint arXiv:1610.05492, 2016.

[Ko18]   Kowsari, K.; Heidarysafa, M.; Brown, D. E.; Meimandi, K. J.; Barnes, L. E.: RMDL: Random Multimodel Deep Learning for Classification. In: Proceedings of the 2nd International Conference on Information System and Data Mining. ACM, Lakeland FL USA, pp. 19–28, 2018, ISBN: 978-1-4503-6354-9, DOI: 10.1145/3206098.3206111, URL: https://dl.acm.org/doi/10.1145/3206098.3206111.

[Kr09]   Krizhevsky, A.: Learning multiple layers of features from tiny images, tech. rep., 2009.

[La23]   Lange, L.; Schneider, M.; Christen, P.; Rahm, E.: Privacy in Practice: Private COVID-19 Detection in X-Ray Images. In: 20th International Conference on Security and Cryptography (SECRYPT 2023). SciTePress, pp. 624–633, 2023, ISBN: 978-989-758-666-8, DOI: 10.5220/0012048100003555, URL: https://doi.org/10.5220/0012048100003555.

[Le98]   Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE 86 (11), pp. 2278–2324, 1998, ISSN: 00189219, DOI: 10.1109/5.726791.

[LW14]   Li, C.; Wang, B.: Fisher linear discriminant analysis. CCIS Northeastern University 6, 2014.

[Ma15]   Martín Abadi; Ashish Agarwal; Paul Barham; Brevdo, E.; Zhifeng Chen; Craig Citro; Greg S. Corrado; Andy Davis; Jeffrey Dean; Matthieu Devin; Sanjay Ghemawat; Ian Goodfellow; Andrew Harp; Geoffrey Irving; Isard, M.; Jia, Y.; Rafal Jozefowicz; Lukasz Kaiser; Manjunath Kudlur; Josh Levenberg; Dandelion Mané; Rajat Monga; Sherry Moore; Derek Murray; Chris Olah; Mike Schuster; Jonathon Shlens; Benoit Steiner; Sutskever, I.; Kunal Talwar; Paul Tucker; Vincent Vanhoucke; Vijay Vasudevan; Fernanda Viégas; Oriol Vinyals; Pete Warden; Martin Wattenberg; Martin Wicke; Yuan Yu; Xiaoqiang Zheng: TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, Software available from tensorflow.org, 2015, URL: https://www.tensorflow.org/.

[Ma21]   Malek Esmaeili, M.; Mironov, I.; Prasad, K.; Shilov, I.; Tramer, F.: Antipodes of label differential privacy: Pate and alibi. Advances in Neural Information Processing Systems 34, pp. 6934–6945, 2021.

[Ne11]   Netzer, Y.; Wang, T.; Coates, A.; Bissacco, A.; Wu, B.; Ng, A. Y.: Reading digits in natural images with unsupervised feature learning. 2011.

[OB05]   O'Brien, J. W.: The JPEG image compression algorithm. APPM-3310 FINAL PROJECT (4), pp. 4–7, 2005.

[Pa18]   Papernot, N.; Song, S.; Mironov, I.; Raghunathan, A.; Talwar, K.; Erlingsson, Ú.: Scalable Private Learning with PATE. In: 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings. OpenReview.net, 2018, URL: https://openreview.net/forum?id=rkZB1XbRZ.

[Po23]   Ponomareva, N.; Hazimeh, H.; Kurakin, A.; Xu, Z.; Denison, C.; McMahan, H. B.; Vassilvitskii, S.; Chien, S.; Thakurta, A.: How to DP-fy ML: A Practical Guide to Machine Learning with Differential Privacy. 2023, DOI: 10.48550/ARXIV.2303.00654, URL: https://arxiv.org/abs/2303.00654.

[PW17]   Perez, L.; Wang, J.: The Effectiveness of Data Augmentation in Image Classification using Deep Learning. 2017, DOI: 10.48550/ARXIV.1712.04621, URL: https://arxiv.org/abs/1712.04621.

[Ra18]   Rahman, M.; Rahman, T.; Laganière, R.; Mohammed, N.: Membership Inference Attack against Differentially Private Deep Learning Model. Trans. Data Priv. 11, pp. 61–79, 2018, URL: https://api.semanticscholar.org/CorpusID:13699042.

[Ra21]   Rahman, T.; Khandakar, A.; Qiblawey, Y.; Tahir, A.; Kiranyaz, S.; Kashem, S. B. A.; Islam, M. T.; Al Maadeed, S.; Zughaier, S. M.; Khan, M. S., et al.: Exploring the effect of image enhancement techniques on COVID-19 detection using chest X-ray images. Computers in biology and medicine 132, p. 104319, 2021.

[Sh17]   Shokri, R.; Stronati, M.; Song, C.; Shmatikov, V.: Membership Inference Attacks Against Machine Learning Models. In: 2017 IEEE Symposium on Security and Privacy (SP). IEEE, San Jose, CA, USA, pp. 3–18, 2017, ISBN: 978-1-5090-5533-3, DOI: 10.1109/SP.2017.41, URL: http://ieeexplore.ieee.org/document/7958568/.

[Sh48]   Shannon, C. E.: A Mathematical Theory of Communication. The Bell System Technical Journal 27, pp. 379–423, 1948.

[SJS06]   Sokolova, M.; Japkowicz, N.; Szpakowicz, S.: Beyond Accuracy, F-Score and ROC: A Family of Discriminant Measures for Performance Evaluation. In (Sattar, A.; Kang, B.-h., eds.): AI 2006: Advances in Artificial Intelligence. Vol. 4304, Lecture Notes in Computer Science, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 1015–1021, 2006, ISBN: 978-3-540-49787-5, DOI: 10.1007/11941439_114, URL: http://link.springer.com/10.1007/11941439_114.

[SP23]   Shamsabadi, A. S.; Papernot, N.: Losing less: A loss for differentially private deep learning. Proceedings on Privacy Enhancing Technologies, 2023.

[Sr14]   Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R.: Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research 15 (56), pp. 1929–1958, 2014.

[To20]   Tonni, S. M.; Vatsalan, D.; Farokhi, F.; Kaafar, D.; Lu, Z.; Tangari, G.: Data and Model Dependencies of Membership Inference Attack. 2020, DOI: 10.48550/ARXIV.2002.06856, URL: https://arxiv.org/abs/2002.06856.

[Tr19]   Truex, S.; Liu, L.; Gursoy, M. E.; Wei, W.; Yu, L.: Effects of Differential Privacy and Data Skewness on Membership Inference Vulnerability. In: 2019 First IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA). IEEE, Los Angeles, CA, USA, pp. 82–91, 2019, ISBN: 978-1-72816-741-1, DOI: 10.1109/TPS-ISA48467.2019.00019, URL: https://ieeexplore.ieee.org/document/9014384/.

[Tr21]   Truex, S.; Liu, L.; Gursoy, M. E.; Yu, L.; Wei, W.: Demystifying Membership Inference Attacks in Machine Learning as a Service. IEEE Transactions on Services Computing 14 (6), pp. 2073–2089, 2021, ISSN: 1939-1374, 2372-0204, DOI: 10.1109/TSC.2019.2897554, URL: https://ieeexplore.ieee.org/document/8634878/.

[Un21]   Uniyal, A.; Naidu, R.; Kotti, S.; Singh, S.; Kenfack, P. J.; Mireshghallah, F.; Trask, A.: DP-SGD vs PATE: Which Has Less Disparate Impact on Model Accuracy? 2021, DOI: 10.48550/ARXIV.2106.12576, URL: https://arxiv.org/abs/2106.12576.

[VGR20]   Vakili, M.; Ghamsari, M.; Rezaei, M.: Performance Analysis and Comparison of Machine and Deep Learning Algorithms for IoT Data Classification. 2020, DOI: 10.48550/ARXIV.2001.09636, URL: https://arxiv.org/abs/2001.09636.

[XBJ21]   Xu, R.; Baracaldo, N.; Joshi, J.: Privacy-Preserving Machine Learning: Methods, Challenges and Directions. 2021, DOI: 10.48550/ARXIV.2108.04417, URL: https://arxiv.org/abs/2108.04417.

[XRV17]   Xiao, H.; Rasul, K.; Vollgraf, R.: Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms. CoRR abs/1708.07747, 2017, arXiv: 1708.07747, URL: http://arxiv.org/abs/1708.07747.

[YW13]   Yu, H.; Winkler, S.: Image complexity and spatial information. In: 2013 Fifth International Workshop on Quality of Multimedia Experience (QoMEX). IEEE, Klagenfurt am Wörthersee, Austria, pp. 12–17, 2013, ISBN: 978-1-4799-0738-0, DOI: 10.1109/QoMEX.2013.6603194, URL: http://ieeexplore.ieee.org/document/6603194/.