Does Learning Stable Features Provide Privacy Benefits for Machine Learning Models?

Divyat Mahajan
Microsoft Research
Bangalore, India

Shruti Tople
Microsoft Research
Cambridge, United Kingdom

Amit Sharma
Microsoft Research
Bangalore, India

Abstract

Privacy attacks such as membership and attribute inference are a serious concern when using machine learning models, and more so when these models are used over data distributions different than their training distribution. As a defense, models that have better generalization properties, including domain generalization (DG) methods that aim to learn stable feature representations across distributions, have been theoretically shown to reduce risk to such privacy attacks. In this work, we investigate the connection between out-of-distribution (OOD) generalization, learning stable features and privacy attacks by performing a rigorous empirical study on two benchmark datasets (Rotated-MNIST and Fashion-MNIST) and a real-world healthcare dataset based on Chest X-ray images. We find that DG methods that rely on learning stable features in theory and indeed learn them in practice provide better privacy guarantees. However, not all state-of-the-art learning algorithms are able to learn the stable features, even though they are optimized for it. On the negative side, we observe that the relationship between OOD accuracy and privacy is not straightforward: a model with high OOD accuracy may also have high privacy risk. Thus, our results indicate the importance of learning stable features to mitigate membership and attribute inference attacks without degrading utility, and motivate the development of better learning algorithms that can learn stable features in practice. For the machine learning community, our work provides novel privacy-based metrics that can be used to measure stability of features learnt by a model.

1 Introduction

Machine learning (ML) models have been shown to be susceptible to *membership inference* attacks, where an attacker can detect whether a data point belongs to a model's training set or not [47]. These attacks have been connected to overfitting [57], and the risk increases when the data points come from a different distribution than the train distribution since it is generally harder for a ML model to generalize to data from a new distribution compared to the train distribution. Having different distributions for inference is quite common in real-world applications. For example, a hospital in one region may train a model and share it to other hospitals.

Outside of differential privacy based defenses that add noise and impact utility of a model [2,10,18,39], recent literature indicates that better generalization of models is an effective way of ensuring membership privacy of training points [51,55,57], such as use of regularization and dropout [37,45]. When data points are not guaranteed to be from the same train distribution, Tople et al. [51] show that causal learning methods [4,33] that aim to learn stable features across distributions alleviate membership privacy risks compared to associational models such as neural networks. Thus, theoretical research suggests that learning stable features avoids overfitting to the train distribution (not just training data), and hence should perform better with respect to privacy.

In another kind of privacy risk, attacks have been shown to learn properties or attributes of inputs that are correlated to the output but not necessary relevant for the prediction task of the model [14, 34, 49]. Such attacks are called *attribute or property inference* attacks. For example, learning the gender attribute of a health record provided as an input to the hospital by simply observing the output predictions. In theory, training models that learn stable features across different distribution of datasets would also defend against leaking sensitive attributes correlated to the output task [51].

In this paper we empirically investigate the theoretical claims connecting out-of-distribution (OOD) generalization, learning stable features and privacy. We define stable (or *causal*) features X_c as those whose relationship with the outcome, $P(Y|X_c)$ remains invariant across different data distributions. To measure the extent to which a learning algorithm learns generalizable features, we consider the accuracy of the trained model on new, unseen data drawn from a different distribution than the train domain (*out-of-distribution* accuracy). To measure stability of the learnt features, we use a recently proposed *mean rank* metric as an additional measure to evalu-

ate whether a model has learned stable features or not [33]. Finally, we connect these two metrics from the machine learning literature to the privacy risk of deploying ML models, by constructing membership inference [45, 47, 50] and attribute inference attacks [14, 34, 49] on these ML models.

Specifically, using image datasets based on Rotated-MNIST [16], Rotated Fashion-MNIST [56] and Chest-Xrays [1, 24, 54], our goal is to check whether better OOD task accuracy leads to lower membership attack scores, and whether the relationship can be explained by stability of features learnt by the ML models. We consider two types of ML models: standard empirical risk minimizers, and state-of-the-art domain generalization learning algorithms that claim to learn stable features for OOD generalization [4, 33, 41]. We do not compare against models trained with defenses for membership privacy such as differential privacy [2, 10, 18, 39] or adversarial regularization [37] since our goal is to understand whether learning algorithms optimized purely with OOD generalization inherently exhibit better privacy guarantees (without degrading utility or accuracy).

On membership inference, we find that stability of features is a more reliable metric for privacy than OOD accuracy. Overall, methods that are able to learn stable features are more robust to membership inference attacks on the three datasets. Thus, we confirm the theoretical result that models that capture stable features are robust to membership inference (MI) attacks. That said, a limitation is that state-of-the-art implementations do not always capture the true, stable features. As a result, learning algorithms based on capturing stable features like Invariant Risk Minimization (IRM) [4] are susceptible to membership inference. On the relationship between OOD accuracy and privacy, we find that OOD accuracy does not provide a good proxy for membership privacy—simple methods that enforce same-class inputs to have similar feature representations may achieve high OOD accuracy, but are still vulnerable to membership attacks.

To evaluate attribute inference, we consider two types of attributes: the domain associated with each input (e.g., gender or age when data is collected from different demographic groups), and a correlated attribute that is common to data from all domains. Here we find that stability of features matters even more. Algorithms like Common-specific low-rank decomposition (CSD) [41] perform well on membership inference robustness, but still leak information about the domain. In comparison, algorithms like Perfect-Match [22, 33] that ideally capture only the stable features, perform well on both membership inference and attribute inference.

Our findings have implications for both ML generalization and privacy communities. For ML privacy, we show that learning stable features can be the key to achieving privacy-robust models that achieve high accuracy too on unseen domains (without compromising on utility). For the ML generalization community, we introduce two novel metrics based on privacy attacks (membership and attribute inference) that can be used to evaluate the stability of features learnt by any ML model. Overall, our results provide evidence towards the development of better, stable learning algorithms that can provide both high accuracy and privacy.

Contributions. Our main contributions are as follows:

- Capturing stable features can be a good way for improving membership and attribute privacy without compromising on utility, but current ML models do not always capture those stable features.
- Better OOD accuracy is not a good proxy for robustness to either membership or attribute inference attacks.
- Among ML algorithms, a method based on self-augmentations (Perfect-Match [22, 33]) provides the best privacy robustness. Apart from PerfectMatch, other DG algorithms present a mixed story: they perform better than standard empirical risk minimization (ERM) when there is substantial spurious correlation in the data, otherwise ERM also does comparably well.

2 Background & Problem Statement

Our work bridges two streams of work: privacy attacks on machine learning models, and training methods based on invariant, stable features that can generalize to unseen distributions.

2.1 Machine Learning Privacy and Generalization

Machine learning models have been shown to be susceptible to several privacy attack that target the inputs or the model parameters, such as membership inference, attribute inference, model stealing [52] and model inversion [11]. In this paper we mainly focus on membership and attribute inference attacks.

Membership Inference. In a membership inference attack on an ML model, the adversary's goal is to infer whether a data point belongs to the training set of the model. Remarkably, with only a black-box access to the ML model (and no access to the training data or the model parameters), attacks have been created that can guess the membership of an input with high accuracy [45, 47]. Shokri et al. introduced membership inference attacks in their seminal paper, [47] demonstrating the risks of inferring presence or absence of a data-point in the training set. They proposed the use of shadow models to generate features and train the membership classifier. Salem et al. remove the requirement of training shadow models and propose to use only the prediction entropy when a subset of the training and test dataset are known [45]. Since then, the prediction entropy metric has been commonly used for membership attack evaluation [7, 48, 57, 58], and the metric

was further improved recently by Song et al. [50]. When white-box access is assumed, the accuracy of the adversary has shown to go up even higher [38].

Such attacks have prompted studies on why deep learning models are susceptible to revealing information on their training set membership. A common reason cited is overfitting to the training dataset, or low capability of the model to generalized to unseen examples from the same data distribution [49, 57]. Under bad generalization properties, the confidence in the prediction values and the resultant accuracy will be lower on a non-training point than a training point, and this difference can be used to develop a membership inference attack.

Attribute Inference. In an attribute or property inference attack, an adversary's goal is to infer the value of a specific property (or input feature) based on the output score of a ML model [5, 14, 49]. Knowing only the black-box score of a ML model, attacks have been constructed that can detect the value of a given attribute with high accuracy [44], thus revealing sensitive data about the input data point. Attribute inference attack is successful whenever the given attribute influences the prediction score of a model in a significant way. It is not always undesired: a good ML model should assign influence to relevant features. However, if the feature is sensitive (and not necessarily required for a correct prediction), we prefer that the model ignores that feature. Note that simply ignoring the specific attribute during model training is not enough [59], since there may be other correlated attributes. Instead, we need that the model output scores are statistically independent of the attribute's value. Note that property inference attacks we consider are different than attribute inference attack shown by Fredrikson et al. where the attacker tries to learn one unknown feature with the access to all other features [12].

2.2 Out-of-Distribution Generalization and Learning Stable Features

Membership attacks can be exacerbated in the real world where there is no guarantee that the input examples will belong to the same distribution as the training dataset. For example, consider a model trained on data from a hospital that can be evaluated on data from a different hospital. Membership inference attack now compares the model's output on the training data compared to another datapoint that does not even share the training data's distribution. From a machine learning perspective, generalization is harder. A model needs to generalize to a different unseen distribution, which standard training methods do not perform well on [4, 20, 21, 40].

Therefore, there is growing interest in the task of building domain generalization algorithms that can perform well on unseen data domains or distributions. In a standard domain generalization task, a learning algorithm is given access to data from multiple training domains and the goal is to build a model that will have high accuracy on data from the same domains, but also from unseen domains. Formally, different domains correspond to different feature distributions $P(X)$ (*covariate shift*) and/or different conditional distributions $P(Y|X)$ (*concept drift*). The unseen domains are restricted in a reasonable way to avoid evaluating on data distributions that can be completely unrelated: for example, domains might be different locations or sites from which data has collected, different views or lighting conditions for photos, etc. Formally, we assume that all domains share some stable features X_C that *cause* the output label, for which the ideal function $P(Y|X_C)$ remains invariant across all training and test domains.

Based on the above discussion, domain generalization (DG) algorithms that aim to learn stable features can provide effective defense against membership inference attacks, even when the inference time data may come from a different distribution. This can be an additional advantage of DG algorithms, which are usually motivated with the goal of better out-of-distribution accuracy. Not all DG algorithms, however, aim to learn stable features. Early work in DG focused on proposing regularizers to the standard training objective of minimizing loss over all training domains [9, 35]. For instance, a simple regularizer-based method is to ensure that same-class inputs from different domains have the same output representation from the model. This helps to align inputs from different domains, but does not learn stable features since inputs from the same class can differ even within a single domain, and do not necessarily share the same stable (causal) features. Other methods [17, 30, 32, 36] have proposed learning a representation $\Phi(x)$ such that the distribution of the representation learnt $P(\Phi(x)$ or $P(\Phi(x)|Y)$ remains the same across domains, but they suffer from similar issues on being reliant on the class label.

State-of-the-art DG algorithms do aim to learn stable features. Improving upon the class-based regularization above, matching methods based on causal inference also been proposed that aim to match representations of inputs that share the base object [33]. The base object refers to the same image or object that may be tranformed to obtain a different input data point in each domain (typically an object's input is observed only in a single domain). Base objects, by definition, share the causal features. The invariant risk minimization (IRM) [4] algorithm learns a representation such that the ideal classifier built on this representation is the same across domains. Corresponding directly to the definition of stable features ($P(Y|\phi(X))$ is invariant across domains), the method aims to learn stable features. Another method [41] aims to separate out input features into two parts such that one of them has common feature weights across domains, and uses only those commmon (stable) features for prediction.

For attribute inference, the objective is different: the learnt model should not be based on sensitive attributes to prevent leaking information about the attribute even if the model generalizes well. In some cases, it is unavoidable since the sensitive attribute may contain useful signal for classifica-

tion, e.g., a person's genetic data for predicting severity of a disease. But in many other settings, such as classifying pneumonia based on chest x-rays [1, 24, 54], sensitive features like gender should ideally play no role in the model's prediction (even though it may be correlated with the observed label). Therefore, to the extent possible, we prefer models that do not make use of sensitive or correlated features so that they do not leak information about those features' values. This goal also aligns with the goal in the domain generalization task of learning stable or causal features for building a classifier, that provide the maximum generalizability without making use of correlated features whose effect can change with diffferent data distributions. In this work, we will consider two types of attribute inference attacks: the first uses the domain as a sensitive attribute, and the second uses a separate attribute whose distribution does not depend on the domain.

2.3 Connection between Privacy and Learning Stable Features?

The connection of membership inference and attribute inference to generalization and stable features indicates that models based on stable features should be more robust to these privacy attacks than standard machine learning models. Theoretically, it has been shown that models that learn stable or causal features are more differentially private than standard loss minimization models (*associational* models) and hence are more robust against membership inference attacks [51]. However, the result assumes perfect knowledge of causal features that is not usually available in real world classification tasks.

In this work, we empirically evaluate the connection between privacy and learning stable features, and between privacy and out-of-distribution accuracy. While past work has empirically studied the connection between same-distribution generalization and privacy [47, 55, 57], our work is the first that analyzes multiple data distributions. Specifically, we evaluate membership and attribute inference attack accuracies of different DG methods based on stable features, and compare them to attack accuracies of standard empirical risk minimizer (ERM) algorithm and other baseline DG methods. We also study the connection between out-of-distribution accuracy and privacy. Specifically, we ask the following questions, assuming that attribute inference attacks are on the correlational features:

- Does higher out-of-distribution generalization lead to better membership and attribute privacy?
- Do methods that learn stable features achieve better membership and attribute privacy?

3 Preliminaries and Metrics

We evaluate the privacy and generalization properties of a training method using four different metrics: membership inference accuracy, attribute inference accuracy, out-of-domain task accuracy, and stability of features as measured by the mean rank metric for learned representations.

3.1 Membership Inference (MI) Attacks

Several methods have been proposed in the literature to compute MI attack accuracy based on the threat-model i.e., black-box or white-box and computational power of the attacker. All these methods identify the boundary that helps to distinguish between members and non-members. As our goal is to use MI attack accuracy as a measure for privacy, we select three different methods that are popularly used in the literature.

For all the different attack techniques, we first create an attack train and attack test dataset using a subset of the original training and test dataset used by the ML model. Thus, our attack accuracy results are an upper bound for all the training techniques. We then infer the parameters (output probabilities) required for the privacy attack to classify members from non members on the attack train dataset, and evaluate its performance on attack test dataset.

Classifier-based attack [47] The classifier based attack was first proposed in the seminal work by Shokri et al. [47] and has been used commonly in several works to demonstrate membership inference attacks [44, 45, 49, 59]. In this method, the adversary trains a classifier, often called as a meta-classifier or an attack classifier to predict member vs. non-member. The input for this classifier is generated by training several shadow models that are trained in a similar fashion as the target model. The output probability vector of several known members and non-members from these shadow models is provided as input for training the classifier.

Instead of training shadow models, we assume access to a subset of training dataset and directly query the target model to generate the input features for the attack classifier. The upper bound works for us since our goal is to perform a relative comparison of different training technqiues using the MI attack.

Loss-based attack [57] Instead of training shadown models, Yeom et al. proposed MI attack that relies on the loss of the target model. The attacker observes the loss values of a few samples and identifies a threshold that distinguishes members from non-members. The intuition is that training data points will have a lower loss value as compared to test data points. This attack is computationally cheap and the attacker does not need to train shadow models or an attack classifier. However, the attack assumes access to the loss values of the target model i.e., it requires white-box access to the model.

Prediction Entropy-based attack [45, 50]. The last method we use was first proposed in Salem et al. [45] and recently improved by Song et al. [50]. This method uses the prediction entropy computed from the probability vector to identify a threshold that distinguishes between members and non-members. For each data point, we predict members vs non members by checking whether the entropy score is below a certain threshold or not. This method improves on both the above method as it is computationally cheap i.e., does not require training a classifier and works in a black-box setting.

3.2 Attribute (Property) Inference (AI) Attacks

Recently, several attacks have demonstrated the machine learning models can leak attribute values or properties of input data that are simply correlated but not relevant for the desired prediction task [14, 34, 49, 59]. Song et al. show that this because of overlearning the model which then utilizes correlated features for the prediction task instead of stable features. Such correlated features may be sensitive but not directly a predictor of the final task. Zhang et al. [59] show that such correlated attributes can be leaked even if they are removed or ignored during the training because of their correlation to other features.

We use a classifier-based attribute inference attack as our second metric to evaluate the privacy of different training techniques that aim for domain generalization. The attack is similar to the MI classifier attack, where we query a subset data points to the target model and use their prediction probability vector as input feature for the attack classifier. The ground truth is the value of the sensitive attribute for the given input. The classifier is trained to predict the attribute value for a given input feature of probability vector. This attack works in a black-box setting.

With the goal to evaluate domain generalization training techniques, we consider two scenarios in our attribute inference attack. In the first case, we consider the domain itself a as a sensitive attribute and for the second case, we introduce a synthetic attribute that is equally present in all the domains and has a correlation with the output class.

3.3 Out-of-Domain Accuracy

To understand the generalization ability of a training technique, we use the accuracy as a measure when computed on inputs that are generated from domains that are *not* seen during training. This is different than standard test accuracy measure where often the validation and test data have the same distribution. Since our goal is to understand the connection between domain generalization techniques and privacy, we select out-of-domain accuracy as one of our evaluation metrics.

3.4 Measuring Stable Features

Measuring stability of learnt features is a hard task, since the ground-truth stable features are unknown for a given classification task. For example, in a MNIST task to classify the digit corresponding to an input image, the shape of the digit can be considered as the stable (or causal) feature whereeas its color or rotation are not stable features. Even if stable features are known, in image datasets, they are typically high-level features (such as shape) that themselves need to be learnt from data. Therefore, verifying stable features directly is a non-trivial task.

Instead, we use the property described by causal domain generalization work [19, 33] that input images that share the base object have the same stable (or causal) features. Thus, if we can select pairs of inputs with the same base object, then they should share the same causal features. Such a pair is known as a **perfect match**. Note that a base object refers to the same semantic input such as a person or a handwritten digit. Input images may consist of the same person in different views or the same handwritten digit in different colors or rotations, but their base object remains the same. Note that there is a many-to-one relationship between an object and its class label. Each class label consists of many objects, which in turn consist of many input images that are differentiated by certain non-stable features like view or rotation or noise.

We use the above property to construct a ranking metric for any ML model. We consider the last layer (logit) representation of the ML model and use a standard distance metric to represent similarity between any two inputs by comparing their representation distances. Then, for each image in the dataset for which a ground-truth perfect match is known, we compute the rank of its ground-truth perfect match based on the ML model's representation. Lower rank is better: a rank of 1 indicates that the learnt feature representation accurately captures the perfect matches, and thus is consistent with a representation based only on stable features (at least on the known perfect matches).

Formally, the mean rank metrics is computed as follows. For the matches (j, k) as per the ground-truth perfect match strategy Ω, compute the mean rank for the data point j w.r.t the learnt match strategy $\Omega^{'}$ i.e. $S_{\Omega'}(j)$

$$\frac{\sum_{\Omega(j,k)=1;d\neq d'} Rank[k \in S_{\Omega'}(j)]}{\sum_{\Omega(j,k)=1;d\neq d'} 1} \tag{1}$$

The efficacy of the mean rank metric depends on the ground-truth perfect matches. For simulated datasets like Rotated-MNIST and Fashion-MNIST, we know the ground-truth matches for each input and use that to evaluate the mean rank metric (the same matches are not provided to the DG methods, except to the ideal Perfect-Match method). For the real-world Chest X-rays dataset, we instead construct self-augmentations for each image by utilizing common transformations like flip from the computer vision literature. We

use these self-augmentations as perfect matches (again, these matches are not provided to the DG methods except for the Hybrid method, which has access to them).

4 Experimental Setup

For our empirical study, we select different state-of-the-art training algorithms for domain generalization and popular image datasets used to evaluate the accuracy of these algorithms.

4.1 OOD Training Methods and ERM

There are several algorithms proposed for domain generalization [4, 9, 13, 29–31, 36, 41]. For our experiments, we use recently proposed methods that aim to learn stable features in their training algorithm, have state-of-the-art OOD accuracy on benchmark datasets, and have code available online. For all methods and the respective loss equations, we use S to denote the set of source domains, N_d as the total number of samples for domain d, f as the classification model, L_d as the classification loss, and x, y to represent the data point and its corresponding true class label.

ERM-Baseline: As our baseline, we use the empirical risk minimization approach to train the model, which minimizes the empirical average of loss over training data points. $\sum_{d\sim S, i\sim N_d} L_d(f(x_i), y_i)$

It treats the data from different domains as i.i.d and simply augments them. This may lead to issues with OOD generalization [4, 40] as we need to learn representations that are robust to the changes in the domains. Hence, a variety of approaches (described below) augment the empirical average loss with regularizers to learn domain invariant models.

Random-Match [9, 33, 35]. Random-Match matches pairs of same-class data points randomly across domains to regularize the model. The idea behind matching across domains is to learn a representations that is invariant to the changes in the source domains, which may lead to better generalization performance. The training loss objective is given by,

$$\sum_{d\sim S, i\sim N_d} L_d(h\circ\phi(x_i), y_i) + \lambda * \sum_{\Omega(j,k)=1 | j\sim N_d, k\sim N_{d'}} Dist(\phi(x_j), \phi(x_k)) \tag{2}$$

where ϕ represents some layer of the network $f = h\circ\phi$, Ω represents the match function used to randomly pair the data points across the different domains. This may not necessarily enforce learning stable features.

CSD [41]. Common-Specific Low-Rank Decomposition (CSD) leads to effective OOD generalization by separating the domain specific and domain invariant parameters, and utilizes the domain invariant parameters for reliable prediction on OOD data. It decomposes the model's final classification layer parameters w as $w = w_s + W * \gamma$, where W represents the k-rank decomposition matrix, w_s represents the domain invariant parameters and γ represent the k domain specific parameters. It optimizes empirical average loss with both the domain invariant and domain specific parameters, along with an orthonormality regularizer that aims to make w_s orthogonal to the decomposition matrix W. Please refer to Algorithm 1 in their paper [41] for more details.

IRM [4]. Invariant Risk Minimization (IRM) aims to learn invariant predictors that simultaneously achieve optimal empirical risk on all the data domains. It minimizes the empirical average loss, and regularizes the model by the norm of gradient of the loss at each source domain as follows:

$$\sum_{d\sim S, i\sim N_d} L_d(w\circ\phi(x_i), y_i) + \lambda * \sum_{d\sim S} ||\nabla_{w|w=1.0} \sum_{i\sim N_d} L_d(w\circ\phi(x_i), y_i)||^2 \tag{3}$$

where $f = w\circ\phi$ and λ is a hyper parameter. In practice, ϕ is taken to be the final layer of the model f, (which makes ϕ and f to be the same). Hence, minimizing the above loss would lead to low norm of the domain specific loss function's gradient and guide the model towards learning an invariant classifier, which is optimal for all the source domains.

MatchDG [33]. The algorithm enforces the same representation for pairs of data points from different domains that share the same causal features. It uses contrastive learning to learn a *matching* function to obtain pairs that share stable causal features between them. The loss function for the method is similar to that of Random-Match (Eq 2), with Ω representing the match function learnt by contrastive loss minimization. Hence, the algorithm consists of two phases; where it learns the matching function Ω in the first phase, and then minimizes the loss function in Eq 2 during the second phase to learn the final model. Please refer to the Algorithm 1 in Mahajan et al. [33] for more details.

Perfect-Match [22, 33]. Finally, we use an algorithm that can be considered to learn *true stable* features for given data, since it relies on knowledge of true base object for a subset of images (and thus guaranteed shared causal features between them). This approach again has a similar formulation to Eq 2, where the match function Ω is satisfied for data points from different domains that share the same base causal object. Hence, it aims to learn similar representations for two data points that only differ in terms of the domain specific attributes.

Hybrid [33]. Perfect matches as explained above are often unobserved but given through Oracle access. However, in

real datasets, augmentations can also provide perfect matches, leading to the *Hybrid* approach using both the MatchDG and augmented Perfect-Match. It learns two match functions, one on the different source domains as per MatchDG, and the other on the augmented domains using Perfect-Match.

Note that Perfect-Match is an ideal training algorithm that assumes knowledge of ground-truth matches across domains, and therefore cannot be applied in real-world settings. In contrast, the Hybrid algorithm depends on creating matches of the same base object using self-augmentations and can be used practically whenever augmentations are easy to create (such as in image datasets). In the experiments that follow, we use the PerfectMatch algorithm for the simulated Rotated-MNIST and Fashion-MNIST datasets, where it should be considered as an ideal method. For the real-world Chest X-rays dataset, we use the practical Hybrid algorithm since we have no knowledge about the true perfect matches.

4.2 Datasets

Rotated-MNIST. This dataset is built upon the MNIST handwritten digits, where we create multiple domains by rotating each digit with the following angles 0°, 15°, 30°, 45°, 60°, 75°, 90° [16]. Angles from 15° to 75° constitute the source domains for training and the angles 0° and 90° are the unseen test domains. Here, the different images across domains share the same base causal object and provide access to true matches required for Perfect-Match approach. Following [41], we sample 2000 data points for each domain, which contribute to the train/test dataset and use ResNet-18 models for classification. To create validation set, for each domain we sample additional data points and makes its size as 20% of the training size.

Fashion-MNIST. This dataset [56] has the same structure as MNIST, where it replaces the handwritten digits by fashion items. We follow the same setup as above for Rotated-MNIST, except we sample $10,000$ data points per domain.

ChestXRay. To evaluate on a more practical scenario, we use the dataset of Chest X-rays images from [33], that comprises of data from different hospital systems: NIH [54], CheXPert [24] and RSNA [1]. The task is to train a classifier that predicts whether a patient suffers from Pneumonia or not. To simplify the analysis, we remove the class imbalance from each of the three domains. For each domain, we sample all the images corresponding to class (1), divide them as 30% for test set and the rest for the train set of that domain. Also, we create a validation set for each domain by further sampling 25% of the training set. Then for all the various splits on each domain, we randomly sample images corresponding to class (0) to ensure to class imbalance.

We use the domains (NIH, ChexPert) for training and the unseen domain (RSNA) for evaluation (details in Table 2). Note that this dataset contains spurious correlation in the source domains due to a vertical translation that shifts all the data points in source domains corresponding to class label 0 downwards. This creates a downward shift in the position of lungs for the images with class (0) as compared to the image with class (1), which creates the spurious feature as the difference in the position of lungs. No such spurious correlation is present in the target domain. Thus, the setup penalizes models that rely on spurious correlations for making predictions, as they would get high accuracy on the source domains, while they would fail on the target domains as no such translation exists in the case of test images.

4.3 Implementation Details

We follow the same procedure as in [33] for the OOD training and evaluation of methods. To summarize, we use the model ResNet-18 (no pre-training) for the Rotated-MNIST and Fashion-MNIST dataset, and we use pre -trained DenseNet-121 for the ChestXRay dataset. For the matching based methods (Random-Match, MatchDG, Perfect-Match), we use the final classification layer of the network as ϕ and for the matching loss regularizer (Eq 2). For all the methods across datasets, we use Cross Entropy for the classification loss (L_d), and use SGD to optimize the loss. Also, we use the data from the source domains for validation and never expose the model to any data from the target domains while training. The details regarding the train/validation/test splits for each dataset are described in the section 4.2 and Table 1.

Source code. Our source code is available at the link in the footnote [1]. All the datasets used in our experiments are publicly available to download.

Membership Inference (MI) Attacks. For implementing MI attacks, we first create the attack-train and attack-test dataset from the original train and test dataset for the ML model. We first sample N number of data points (N: $2,00$ for Rotated-MNIST, $10,000$ for Fashion-MNIST, N: $1,000$ for ChestXRay) from both the original train and test dataset to create the attack-train dataset. Similarly, we sample an additional set of N data points from original train, test dataset to create the attack-test dataset.

For the Classifier-based attack, we label the data points from the training dataset as members, and the data points from the test dataset as non members. We then train a 2-layer fully-connected network (with hidden dimensions 8, 4 respectively) to distinguish members from non-members for all the models and datasets. We use Adam Optimizer with learning rate 0.001, batch size 64 and 5000 steps / 80 epochs.

[1] https://anonymous.4open.science/r/3cd41059-40a6-4fba-a966-d1c05c6f8573/

Table 1: Dataset details

Dataset	Total Classes	Total Domains	Source Domains	Target Domains	Samples per Domain
Rotated-MNIST	10	7	15°, 30°, 45°, 60°, 75°	0°, 90°	2000
Fashion-MNIST	10	7	15°, 30°, 45°, 60°, 75°	0°, 90°	10000
ChestXray	7	3	NIH, ChexPert	RSNA	Table 2

Table 2: Train/Val/Test splits for the ChestXRay dataset.

Domain	Training Set	Validation Set	Test Set
NIH	800	200	430
ChexPert	2618	654	1402
RSNA	3367	841	1803

For the Loss-based and Entropy-based attack, we first identify the threshold τ as the max loss/entropy score among all the members in the attack-train dataset, and then scale the threshold by an integer value s chosen randomly from the grid $(1, \tau)$. We then evaluate the performance for the different threshold obtained by scaling, and choose the final threshold as the one that gives the highest classification accuracy on the attack-train dataset. We use the final theshold to evaluate performance on the attack-test dataset. We sample 10 values at random for the scaling factor s for our experiments across datasets. Also, if the initial threshold τ is less than 10 (which would restrict from sampling 10 distinct integers from the grid defined above), we update the grid to be $(1, 20)$ from $(1, \tau)$.

Attribute Inference (AI) Attacks. We use all the source and target domains, and sample data points from their training/test set to create the attack-train/attack-test dataset respectively. We then label data points in the attack-train and attack-test dataset based on the attribute, and then train a classifier to discriminate among different attributes. We follow the same procedure for training the classifier as defined for the Classifier-based attack above.

5 Evaluation Results

We perform our evaluation with the following goals:

- To compare the MI attack accuracy of ERM and DG training methods on Rotated-MNIST, Fashion-MNIST and ChestXray,
- To understand the relation between learning stable features, OOD generalization accuracy (train-test) gap and MI privacy,
- To compare attribute inference attack accuracy of ERM and DG training methods.

5.1 Summary of Key Results

We summarize our observations across all datasets and training methods in Table 3 and show the connection between learning stable features in theory and practice, high OOD generalization and better MI privacy. Our key findings are as follows:

- Training methods that rely on learning stable features in theory and exhibit that in practice consistently provide better privacy guarantees e.g., PerfectMatch/Hybrid and MatchDG. This shows that **learning stable features is a right way forward to ensure membership privacy without degrading utility**.
- Methods that exhibit low OOD generalization error (high OOD test accuracy) but do not learn stable features to achieve the high accuracy fail at mitigating privacy risk e.g., Random-Match on Rotated-MNIST. Similarly, methods that exhibit high OOD generalization error but learn stable features are indeed better at privacy e.g., IRM on Rotated-MNIST and Fashion-MNIST. **OOD generalization error is not a reliable metric for MI privacy risk.**
- Methods that rely on learning stable features in theory may not always learn them in practice e.g., CSD and IRM for ChestXray. Similarly, methods like ERM that do not enforce learning stable features in theory can indeed learn them in practice for some datasets such as Rotated-MNIST and Fashion-MNIST. It is important that domain generalization training techniques are evaluated on real-world datasets like ChestXray than toy datasets such as variations of MNIST.

Our results show that capturing stable features can be a good way for improving MIA without compromising on utility, but current ML models do not always capture those stable features. Our results also show the inadequacy of OOD accuracy as a metric for evaluating domain generalization algorithms: when ground-truth stable features are not available, membership privacy can be a viable metric to evaluate algorithms for learning stable features.

5.2 Membership Inference Attacks

We perform MI attacks on our benchmark datasets when trained using different training algorithms mentioned in Section 4.1. Figure 1 (center) shows the MI attack accuracy re-

Table 3: Connection between learning stable representation, high OOD accuracy and better membership privacy.● denotes the training method satisfies the metric, ○ means it does not.

Metrics	Dataset	ERM	Random-Match	IRM	CSD	MatchDG	Perfect-Match/ Hybrid
Stable Features (Theory)		○	○	●	●	●	●
Stable Features (Practice)	Rotated-MNIST	●	○	●	●	●	●
	Fashion-MNIST	●	○	●	●	●	●
	ChestXray	○	○	○	○	●	●
Out-of-Domain Accuracy	Rotated-MNIST	○	●	○	○	●	●
	Fashion-MNIST	○	○	○	○	●	●
	ChestXray	○	○	○	●	●	●
MI Attack Accuray	Rotated-MNIST	●	○	●	●	●	●
	Fashion-MNIST	●	●	●	●	●	●
	ChestXray	○	○	○	○	●	●

sults on all our datasets. In addition to the attack accuracy, we show training and OOD test accuracy in Figure 1 (left) to understand whether MI attack accuracy has a direct relationship to OOD generalization (train-test gap [47]). Finally, to confirm the connection between MI privacy and learning stable features in practice, Figure 1 (right) shows the mean rank metric which captures the ability of a training method to capture stable features in practice. Comparing the three graphs, we present our observations for each of the dataset below.

Rotated-MNIST. Figure 1a shows the results on the Rotated-MNIST dataset. We observe that all the three types of MI attacks results in similar attack accuracy.

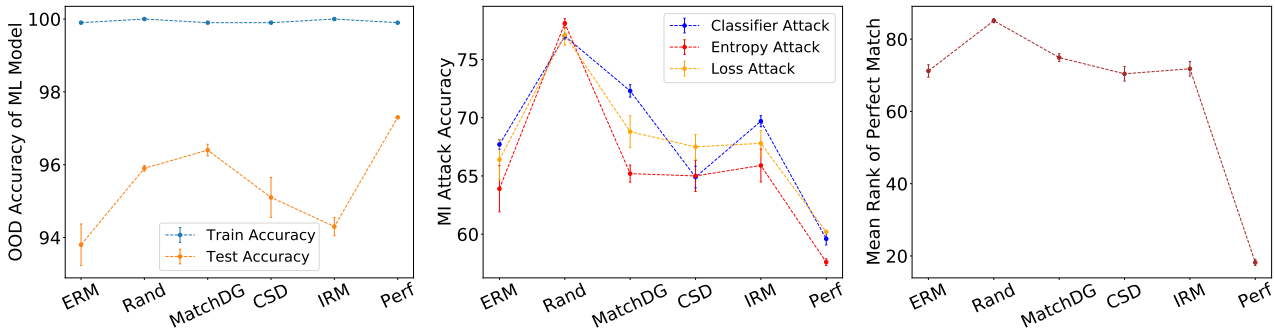
- We observe that the PerfectMatch training algorithm that learns stable features in theory has the lowest mean rank, indicating that it has learnt relatively better stable features in practice as compared to all the other methods. Further, it has the lowest attack accuracy, almost close to random guess baseline of 50% and relatively small OOD generalization gap. This result confirms the theoretical understanding that learning stable features provides inherent privacy guarantees during training.

- MatchDG, CSD, and IRM methods that also enforce learning stable features in theory exhibit higher mean rank than PerfectMatch. They obtain almost similar mean ranks between them. This explains why they have similar MI attack accuracy in practice even though IRM has a higher OOD generalization gap (i.e, lower test accuracy) as compared to CSD and MatchDG. These results indicate that that privacy of a model should not be explained using only its generalization accuracy but one should take into consideration the type of features that the model uses to achieve that accuracy.

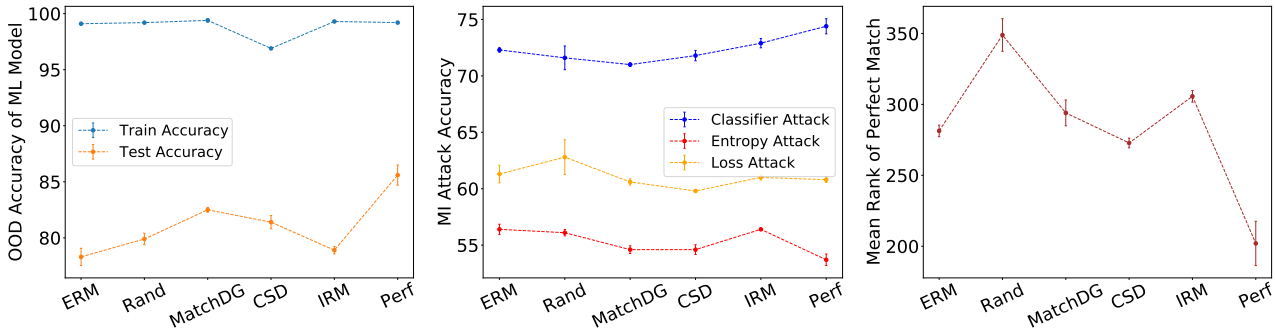
- We observe a similar phenomena with RandomMatch results. This training algorithm does not rely on learning stable features in theory and consequently has the highest mean rank, confirming that it does not learn stable features in practice. On Rotated MNIST and Chest Xray datasets, the 80% MI attack accuracy for this algorithm (compared to $< 70\%$ for IRM and other stable feature-learning algorithms) confirms the theoretical claim that learning stable features is essential to improve privacy of a model. Note that the OOD generalization gap for RandomMatch is smaller than IRM which is based on learning stable features. A smaller generalization gap but a higher MI privacy attack accuracy provides more evidence to our claim that OOD accuracy is not the right metric to understand the privacy guarantees of a model. A smaller generalization gap does not always results in smaller attack accuracy. A possible reason is that the training algorithm may learn some correlated features that does not affect accuracy, but introduces enough variation in the scores of the train and test input points that can be exploited by an adversary.

- For ERM, although it does not enforce learning stable features in theory, we observe that it ends up learning them in practice and hence provide comparable MI privacy guarantees to MatchDG, CSD and IRM that use learning stable features as a part of their training objective.

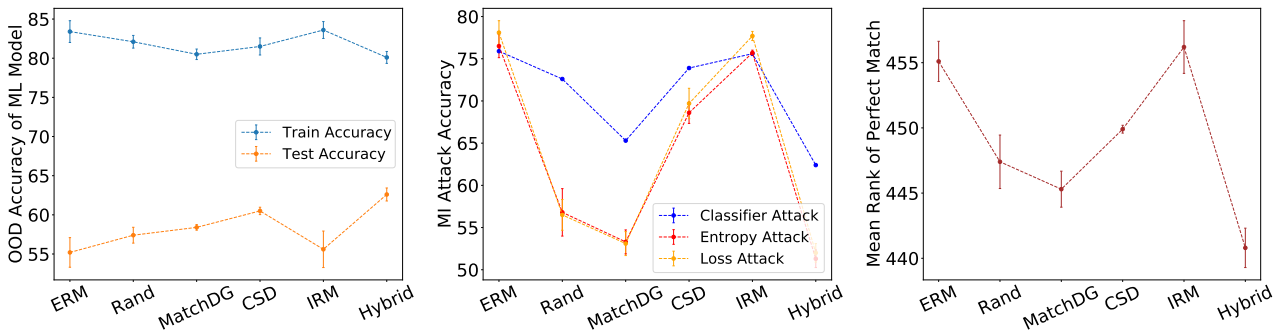
Fashion-MNIST. Figure 1b shows the result for OOD accuracy, MI attack accuracy and mean rank. In contrast to



(a) Rotated MNIST results, left: OOD Accuracy (higher is better), center: MI Attack Accuracy and right: Mean Rank (lower is better)



(b) Fashion MNIST, left: OOD Accuracy (higher is better), center: MI Attack Accuracy and right: Mean Rank (lower is better)



(c) ChestXRay evaluation, left: OOD Accuracy (higher is better), center: MI Attack Accuracy and right: Mean Rank (lower is better)

Figure 1: OOD generalization and MI privacy attacks on Rotated MNIST, Fashion MNIST and ChestXRay datasets.

Rotated-MNIST, we observe that the three MI attack methods exhibit a wide range in obtained attack accuracy. That said, betweeen different DG algorithms, they do exhibit a trend similar to each other, with the loss-based and prediction-entropy attacks achieving higher mutual agreement than the classifier-based attack. Thus, we use the trend from the majority agreement to make our observations.

- For Fashion-MNIST, we do not find a large difference between the MI attack accuracy for different training algorithms. Overall, the relative trend among different learning algorithms for prediction-entropy and loss-based attacks does correlate well with the mean rank metric. Two training algorithms (MatchDG, CSD, PerfectMatch) that enforce learning stable features in theory indeed have relatively low mean rank, and also have low attack accuracy. However, the differences in attack accuracy are quite small (and some are not statistically significant), so we refrain from interpreting differences between individual algorithms.

- For ERM, we observe similar results as that for Rotated-MNIST. It learns stable features in practice (without enforcing that as a learning objective during training) and hence exhibits better privacy guarantees. Note that ERM suffers from the inability to generalize similar to IRM which should not be considered as an indicator of privacy risk.

Table 4: Attribute Attack Accuracy for different datasets with domains as an attribute.

Dataset	Random Guess	ERM	Random-Match	IRM	CSD	MatchDG	Perfect-Match/ Hybrid
Rotated-MNIST	14.3%	27.6 (0.84)	20.4 (0.46)	26.6 (0.71)	25.0 (0.40)	19.3 (0.62)	**16.6 (0.55)**
Fashion-MNIST	14.3%	26.6 (0.59)	21.8 (0.77)	23.8 (0.86)	26.9 (0.93)	21.9 (0.29)	**21.5 (0.97)**
ChestXray	33.3%	61.8 (1.02)	58.4 (0.58)	63.4 (2.28)	67.8 (3.05)	57.9 (0.58)	**57.1 (0.21)**

ChestXray. Figure 1c shows the results for the ChestXray dataset which mimics a scenario in real-world applications. For ChestXray, we observe the three MI attacks agree on the relative trends among different training methods. The loss-based and entropy-based attack have almost overlapping numbers. We make the following observations:

- Comparing between different learning algorithms, the trend for MI attack accuracy and the Mean Rank metric is similar, yielding an almost identical ranking of algorithms based on the two metrics.
- MatchDG and Hybrid (an alternative for PerfectMatch) have low mean rank indicating that they have learnt stable features and hence have lower MI attack accuracy.
- Training methods such as CSD and IRM that enforce learning stable features in theory are unable to capture them in practice for ChestXray dataset which is shown by the relatively higher mean rank values. They also obtain higher MI attack accuracies.
- Unlike Rotated-MNIST and Fashion-MNIST, ERM is unable to learn stable features in practice for the ChestXray dataset which is shown by the high mean rank. This explains the high MI attack accuracy of 78%.
- The MI attack accuracy among different training method exhibit significant variations which are not reflected in their OOD accuracies in Figure 1c(left). In contrast, the similar trend of graphs for MI attack accuracy in Figure 1c(center) and mean rank in Figure 1c(right) shows that learning stable features results in better MI privacy. This observation also has an implication for the domain generalization community as they can use MI attack as a metric to measure the ability of a training algorithm to learn stable features.

5.3 Attribute Inference Attacks

State the assumption that Attribute Attacks are only on spurious features, hence learning stable features will help on Attribute Attacks.

In addition to membership inference attack, we perform attribute inference attacks on the same training algorithms and datasets. For attribute inference, we aim to understand if the model leaks attribute that are not relevant for the final task. We perform two experiments: a) where we select domain itself as the attribute to infer and b) where we introduce a domain-agnostic attribute – color that the attacker tries to infer. Note that both attributes, domain and color in each of the experiments is not relevant for the final prediction label in our datasets. We present the results for both settings below.

Domain as a sensitive attribute. In this attack, we consider domain of the data from which it is generated as a sensitive attribute that the attacker is trying to learn. Table 4 shows the attack results across different training methods on all the three datasets. For Rotated-MNIST and Fashion-MNIST, we train the attack classifier to distinguish among all the 7 angles of rotation and hence the baseline accuracy with random guess 14.33% while for ChestXray, we aim to distinguish among 3 domains resulting in a baseline of 33.33%.

Similar to MI attacks, we observe that PerfectMatch/Hybrid approach has the lowest attack accuracy followed by MatchDG. PerfectMatch/Hybrid, having access to perfect matches based on self-augmentations, has access to the pairs of inputs that share the same causal features, and hence obtains both highest OOD accuracy (as we saw earlier) and the lowest AI attack accuracy. However, there is not a one-to-one correspondence between OOD accuracy and AI attack accuracy. CSD obtains one of the highest OOD accuracies on all datasets but its attack accuracy is one of the highest. We suspect that this discrepancy is due to the different objective of CSD algorithm compared to the matching algorithms of PerfectMatch/Hybrid and MatchDG: CSD does not aim to obtain the same representations across domains. Thus, when domains convey sensitive information, it is preferable to employ matching-based methods that can provide both high OOD accuracy and low attribute inference attack accuracy. Unlike in membership inference attacks, the standard ERM training algorithm does not perform well on attribute inference, obtaining one of the highest attack accuracies.

That said, it is somewhat intuitive that DG training approaches that are designed to be domain invariant are able to defend against attribute inference attacks as compared to ERM. Therefore, we present another attack with a domain-agnostic attribute to understand whether DG methods can mitigate such attacks well.

Table 5: Attribute Attack Accuracy for the dataset Rotated-MNIST with color as an attribute. We consider the attribute as binary (0: no color; 1: color), thus the best case AI Attack accuracy would be 50% via random guess.

Metrics	ERM	Random-Match	IRM	CSD	MatchDG	Perfect-Match/ Hybrid
Train Accuracy	100.0 (0.0)	99.9 (0.02)	99.7 (0.09)	100.0 (0.0)	100.0 (0.01)	97.7 (0.34)
Test Accuracy	96.5 (0.13)	96.8 (0.18)	95.9 (0.21)	97.2 (0.02)	**97.5 (0.09)**	96.4 (0.30)
AI Attack Accuracy	98.5 (0.35)	83.0 (1.96)	99.3 (0.10)	95.4 (0.15)	99.7 (0.09)	**70.2 (0.09)**
Test (Permute) Accuracy	28.4 (0.76)	37.6 (3.93)	31.9 (0.37)	42.9 (3.46)	28.8 (1.27)	**94.9 (0.46)**

Domain-agnostic sensitive attribute. For this attack, we introduce a different color as an attribute with 70% probability to each class of the Rotated-MNIST dataset irrespective of the domain to which the image belongs. The target model is trained with this modified dataset using the same training algorithms as before. The attacker's goal is to infer whether a given input is colored or is black and white (a binary task). Table 5 shows the results for train and test accuracy of the target model and the attribute inference attack accuracy. The test set contains a different rotation angle, but the color distribution remains the same wrt. the class labels.

We observe that PerfectMatch has the lowest attack accuracy as compared to all other training algorithms, demonstrating that the ability to learn stable features provides inherent ability to defend against a harder task of attribute inference when the attribute itself is domain-agnostic. The second-best performing algorithm is RandomMatch, which indicates that AI attack accuracy is not directly connected to stability of features learnt since RandomMatch does not aim to learn stable features. Note that the OOD (test) accuracy does not convey much information about the privacy risk through an AI attack: all methods obtain OOD accuracies within a narrow range of 95.9%-97.5%, but the AI attack accuracies range from 70.2% for PerfectMatch to 99.3% for IRM.

We hypothesize that the difference in AI attack accuracies is because of the differing extents to which the ML models utilize color as a feature. To verify the hypothesis, we construct a new, *permuted* test domain where the color of each colored image is permuted randomly to a different value. Thus, the correlation in the training data between color and the class label is no longer present in the test domain. Under such a test domain, we observe larger differences in OOD accuracy that correspond to the AI attack accuracies reported above: PerfectMatch obtains the highest OOD accuracy (94.9%) as compared to other methods that range between 28% to 42%.

The substantially low accuracies for the other DG methods indicates that state-of-the-art DG methods do not always learn the stable, non-correlational features, and more research is needed to introduce such constraints to the DG methods. We also find that PerfectMatch/Hybrid algorithm always performs well, indicating that whenever self-augmentations are available, using those augmentations can lead to better learning of the stable features.

6 Related Works

Empirical study of MI and Privacy attacks. Privacy attacks such as membership inference are gaining attention and several prior work have performed empirical studies trying to understand these attacks with different goals. Jayaraman and Evans [25] and Rahman et al. [42] demonstrate the efficacy of membership inference attacks when models are trained using differential privacy as a defense. Their main goal is to understand how different ε values for differential privacy affect membership inference attack accuracy. Song et al. [50] perform a systematic evaluation using different method for MI attacks and demonstrate the efficacy of defenses such as adversarial regularization [37] and Memguard [26]. In this work, we approach membership inference attacks from a different angle. We aim to evaluate their effectiveness in out-of-domain generalization setting. Moreover, instead of training these methods with additional defense mechanism, our goal is to study whether OOD generalization approaches provide inherent protection against membership privacy without degrading utility.

Generalization and Privacy. Since overfitting of models has been shown to be the primary reason for membership inference [57], recent approaches have shown that better generalization of models aids in mitigating privacy risk [51, 55] without degrading the model utility. Tople et al. [51] show that generalization property of causal learning approaches due to learned stable features have inherent privacy benefits. Wu et al. show that Lipschitz regularization when used gen-

erative adversarial networks (GANs) not only reduces the generalization gap but alleviates membership privacy risk simultaneously [55]. In the light of these results, this work studies whether OOD generalization approaches exhibit any privacy benefits and its connection to the stability of learned representations.

Domain Generalization Approaches. The domain generalization task is to learn a single classifier, given multiple source domains at training time, that generalizes well to unseen target domains [46]. A variety of approaches have been proposed for achieving the same, with many relying on learning similar distribution of features across domains [13,15,23,29–31,36]. Recently, some approaches [3,4] focus on learning invariant classifiers which are simultaneously optimal for all the domains. There are also works that are inspired by causality [4, 17, 32, 33, 40, 43] and take different interpretations towards a causal framework for domain generalization. Meta learning has also inspired many approaches [6, 9, 28] for domain generalization. Furthermore, there are works [46, 53] that generate novel domains at training time to improve generalization of the model. Finally, there are works [8, 27, 41] that take the approach of disentangling the domain specific and domain invariant model parameters, and utilizing the domain invariant parameters to make predictions on the unseen domains.

7 Conclusion

When do ML models exhibit membership inference and attribute inference risk? We provided empirical evidence for this key question in machine learning privacy, suggesting that stability of learnt features is a key driver of robustness to privacy attacks. Our results confirm earlier theoretical work on the importance of stable or causal features for privacy, and show that past metrics like the train-test generalizability gap are not sufficient for characterizing the membership inference risk.

Our work motivates further study into learning stable features, and better metrics for measuring and training ML models for this objective. Better algorithms for learning stable features can be more robust to privacy attacks, without compromising on utility. Similarly, the resultant privacy metrics can prove useful for ML generalization community to evaluate and build ML models that generalize to unseen domains.

References

[1] Kaggle: Rsna pneumonia detection challenge, 2018.

[2] 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.

[3] Kartik Ahuja, Karthikeyan Shanmugam, Kush Varshney, and Amit Dhurandhar. Invariant risk minimization games. *arXiv preprint arXiv:2002.04692*, 2020.

[4] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

[5] Giuseppe Ateniese, Luigi V. Mancini, Angelo Spognardi, Antonio Villani, Domenico Vitali, and Giovanni Felici. Hacking smart machines with smarter ones: How to extract meaningful data from machine learning classifiers. 2015.

[6] Yogesh Balaji, Swami Sankaranarayanan, and Rama Chellappa. Metareg: Towards domain generalization using meta-regularization. In *Advances in Neural Information Processing Systems*, pages 998–1008, 2018.

[7] Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th {USENIX} Security Symposium ({USENIX} Security 19)*, pages 267–284, 2019.

[8] Hal Daumé III, Abhishek Kumar, and Avishek Saha. Frustratingly easy semi-supervised domain adaptation. In *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*, pages 53–59. Association for Computational Linguistics, 2010.

[9] Qi Dou, Daniel Coelho de Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. In *Advances in Neural Information Processing Systems*, pages 6447–6458, 2019.

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

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

[12] 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.

[13] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

[14] Karan Ganju, Qi Wang, Wei Yang, Carl A. Gunter, and Nikita Borisov. Property inference attacks on fully connected neural networks using permutation invariant representations. In *ACM Conference on Computer and Communications Security (CCS)*, 2018.

[15] Muhammad Ghifary, David Balduzzi, W Bastiaan Kleijn, and Mengjie Zhang. Scatter component analysis: A unified framework for domain adaptation and domain generalization. *IEEE transactions on pattern analysis and machine intelligence*, 39(7):1414–1430, 2016.

[16] Muhammad Ghifary, W Bastiaan Kleijn, Mengjie Zhang, and David Balduzzi. Domain generalization for object recognition with multi-task autoencoders. In *Proceedings of the IEEE international conference on computer vision*, pages 2551–2559, 2015.

[17] Mingming Gong, Kun Zhang, Tongliang Liu, Dacheng Tao, Clark Glymour, and Bernhard Schölkopf. Domain adaptation with conditional transferable components. In *International conference on machine learning*, pages 2839–2848, 2016.

[18] Jihun Hamm, Yingjun Cao, and Mikhail Belkin. Learning privately from multiparty data. In *International Conference on Machine Learning*, pages 555–563, 2016.

[19] Christina Heinze-Deml and Nicolai Meinshausen. Conditional variance penalties and domain shift robustness. *arXiv preprint arXiv:1710.11469*, 2019.

[20] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. *arXiv preprint arXiv:2006.16241*, 2020.

[21] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

[22] Dan Hendrycks, Norman Mu, Ekin D Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv preprint arXiv:1912.02781*, 2019.

[23] Shoubo Hu, Kun Zhang, Zhitang Chen, and Laiwan Chan. Domain generalization via multidomain discriminant analysis. In *Uncertainty in artificial intelligence: proceedings of the... conference. Conference on Uncertainty in Artificial Intelligence*, volume 35. NIH Public Access, 2019.

[24] Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 590–597, 2019.

[25] Bargav Jayaraman and David Evans. Evaluating differentially private machine learning in practice. In *28th {USENIX} Security Symposium ({USENIX} Security 19)*, pages 1895–1912, 2019.

[26] 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.

[27] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pages 5542–5550, 2017.

[28] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Learning to generalize: Meta-learning for domain generalization. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[29] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C Kot. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5400–5409, 2018.

[30] Ya Li, Mingming Gong, Xinmei Tian, Tongliang Liu, and Dacheng Tao. Domain generalization via conditional invariant representations. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[31] Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 624–639, 2018.

[32] Sara Magliacane, Thijs van Ommen, Tom Claassen, Stephan Bongers, Philip Versteeg, and Joris M Mooij. Domain adaptation by using causal inference to predict invariant conditional distributions. In *Advances in Neural Information Processing Systems*, pages 10846–10856, 2018.

[33] Divyat Mahajan, Shruti Tople, and Amit Sharma. Domain generalization using causal matching. *arXiv preprint arXiv:2006.07500*, 2020.

[34] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. Exploiting unintended feature leakage in collaborative learning. In *IEEE Symposium on Security and Privacy*, 2019.

[35] Saeid Motiian, Marco Piccirilli, Donald A Adjeroh, and Gianfranco Doretto. Unified deep supervised domain adaptation and generalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5715–5725, 2017.

[36] Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *International Conference on Machine Learning*, pages 10–18, 2013.

[37] 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. ACM, 2018.

[38] Milad Nasr, Reza Shokri, and Amir Houmansadr. Comprehensive privacy analysis of deep learning: Standalone and federated learning under passive and active white-box inference attacks. In *IEEE Symposium on Security and Privacy*, 2019.

[39] Nicolas Papernot, Martín Abadi, Ulfar Erlingsson, Ian Goodfellow, and Kunal Talwar. Semi-supervised knowledge transfer for deep learning from private training data. In *ICLR*, 2017.

[40] Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(5):947–1012, 2016.

[41] Vihari Piratla, Praneeth Netrapalli, and Sunita Sarawagi. Efficient domain generalization via common-specific low-rank decomposition. *Proceedings of the International Conference of Machine Learning (ICML) 2020*, 2020.

[42] Md Atiqur Rahman, Tanzila Rahman, Robert Laganière, Noman Mohammed, and Yang Wang. Membership inference attack against differentially private deep learning model. *Trans. Data Priv.*, 11(1):61–79, 2018.

[43] Mateo Rojas-Carulla, Bernhard Schölkopf, Richard Turner, and Jonas Peters. Invariant models for causal transfer learning. *The Journal of Machine Learning Research*, 19(1):1309–1342, 2018.

[44] Ahmed Salem, Apratim Bhattacharya, Michael Backes, Mario Fritz, and Yang Zhang. Updates-leak: Data set inference and reconstruction attacks in online learning. In *Usenix Security*, 2020.

[45] Ahmed Salem, Yang Zhang, Mathias Humbert, Mario Fritz, and Michael Backes. Ml-leaks: Model and data independent membership inference attacks and defenses on machine learning models. In *NDSS*, 2019.

[46] Shiv Shankar, Vihari Piratla, Soumen Chakrabarti, Siddhartha Chaudhuri, Preethi Jyothi, and Sunita Sarawagi. Generalizing across domains via cross-gradient training. In *International Conference on Learning Representations*, 2018.

[47] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *Security and Privacy (SP), 2017 IEEE Symposium on*, pages 3–18. IEEE, 2017.

[48] Congzheng Song and Ananth Raghunathan. Information leakage in embedding models. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, 2020.

[49] Congzheng Song and Vitaly Shmatikov. Overlearning reveals sensitive attributes. In *ICLR*, 2020.

[50] Liwei Song and Prateek Mittal. Systematic evaluation of privacy risks of machine learning models. *arXiv preprint arXiv:2003.10595*, 2020.

[51] Shruti Tople, Amit Sharma, and Aditya V. Nori. Alleviating privacy attacks via causal learning. In *ICML*, 2020.

[52] Florian Tramèr, Fan Zhang, Ari Juels, Michael K Reiter, and Thomas Ristenpart. Stealing machine learning models via prediction apis. In *25th {USENIX} Security Symposium ({USENIX} Security 16)*, pages 601–618, 2016.

[53] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. In *Advances in Neural Information Processing Systems*, pages 5334–5344, 2018.

[54] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2097–2106, 2017.

[55] Bingzhe Wu, Shiwan Zhao, Chaochao Chen, Haoyang Xu, Li Wang, Xiaolu Zhang, Guangyu Sun, and Jun Zhou. Generalization in generative adversarial networks: A novel perspective from privacy protection. In *Advances in Neural Information Processing Systems*, pages 307–317, 2019.

[56] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

[57] 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.

[58] Santiago Zanella-Béguelin, Lukas Wutschitz, Shruti Tople, Victor Rühle, Andrew Paverd, Olga Ohrimenko, Boris Köpf, and Marc Brockschmidt. Analyzing information leakage of updates to natural language models. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, 2020.

[59] Wanrong Zhang, Shruti Tople, and Olga Ohrimenko. Dataset-level attribute leakage in collaborative learning. *arXiv preprint arXiv:2006.07267*, 2020.