IRIS AperTO

UNIVERSITÀ DEGLI STUDI DI TORINO

AperTO - Archivio Istituzionale Open Access dell'Università di Torino

Disentangling private classes through regularization

This is the author's manuscript

Original Citation:

Availability:

This version is available http://hdl.handle.net/2318/1929450 since 2024-01-11T13:00:56Z

Published version:

DOI:10.1016/j.neucom.2023.126612

Terms of use:

Open Access

Anyone can freely access the full text of works made available as "Open Access". Works made available under a Creative Commons license can be used according to the terms and conditions of said license. Use of all other works requires consent of the right holder (author or publisher) if not exempted from copyright protection by the applicable law.

(Article begins on next page)

12 July 2024

Disentangling private classes through regularization

Enzo Tartaglione[a], Francesca Gennari[b,c,d,1], Victor Quétu[a], Marco Grangetto[d]

[a] *LTCI, Télécom Paris, Institut Polytechnique de Paris, France*
[b] *Mykolas Romeris University, Lithuania*
[c] *University of Bologna, Italy*
[d] *University of Turin, Italy*

Abstract

Deep learning models are nowadays broadly deployed to solve an incredibly large variety of tasks. However, little attention has been devoted to connected legal aspects. In 2016, the European Union approved the General Data Protection Regulation which entered into force in 2018. Its main rationale was to protect the privacy and data protection of its citizens by the way of operating the so-called "Data Economy". As data is the fuel of modern Artificial Intelligence, it is argued that the GDPR can be partly applicable to a series of algorithmic decision-making tasks before a more structured AI Regulation enters into force. In the meantime, AI should not allow undesired information leakage deviating from the purpose for which is created. In this work, we propose DisP, an approach for deep learning models disentangling the information related to some classes we desire to keep private, from the data processed by AI. In particular, DisP is a regularization strategy decorrelating the features belonging to the same private class at training time, hiding the information about private class membership. Our experiments on state-of-the-art deep learning models show the effectiveness of DisP, minimizing the risk of extraction for the classes we desire to keep private.

Keywords: Deep learning, GDPR, regularization, disentangling, artificial neural networks, private classes
2000 MSC: 68T01,

Email address: `enzo.tartaglione@telecom-paris.fr` (Enzo Tartaglione)

[1] Francesca Gennari is with LAST-JD-RIoE consortium, more specifically Mykolas Romeris University - University of Bologna - University of Turin.

2000 MSC: 68P30

1. Introduction

Currently, a significantly large portion of problems is being solved through the deployment of a deep learning model, considered by most as a "universal problem-solving tool" [5]. Their fast and uncontrolled spread, however, requires legal rules that are influencing or are bound to influence the development of Artificial Intelligence (AI). There are two main reasons for this. Firstly, it is of general interest that the algorithmic decision-making systems that are being described and explained further can be of wide application, irrespective of the private or public entity of the user. There are tools, available in the literature, attempting to provide explanations on the AI's decision process, like GradCam [6]. However, these tools are limited to input-output relationships, and do not provide insights on how to open "the black box". Secondly, it would be irresponsible and unethical to promote algorithms that could create damage the privacy of people whose data are used by AI [7]. To this end, what will be succinctly explained is the legal outlook of the European Union on AI. This is done because the EU has already obtained the role of the legal influencer as far as the discipline of data protection is concerned about drafting the General Data Protection Regulation (GDPR) [8] and because it has started since 2018 in trying to create the conditions to regulate the AI. It all started with the Ethical Guidelines of the AI [9], the Assessment list for trustworthy Artificial Intelligence (ALTAI) for self-assessment [10] and finally the proposal for the regulation of the AI [11]. These two issues will be dealt with subsequently. It is important to specify that the following two subsections are not a commentary of the entirety of the two regulations but a focus on the most problematic aspects which could potentially clash with the development of an AI described further.

1.1. The GDPR and the AI

The GDPR is the first regulation (an EU mandatory legislative act) that tries to balance the right to data protection, and, more largely, to privacy in the EU, with the Big Data economy. The points of contact between AI developers and the GDPR are mainly two: the concept of personal data (i) and data processing (ii). Then it is important to understand how the developers of neural networks and/or companies/research institutes are seen according to the GDPR (iii) and how the principles of data protection by design and by

default (iv) could be relevant for the development of trustworthy and ethical AI models.

Personal data. Whenever one works with data that can be referred to people (and the AI creation field makes no exception), lawyers but especially innovators must follow the GDPR not only to check on compliance but also to ensure that the AI does not infringe on people's privacy and data protection, which in the EU are considered fundamental rights and therefore are always to be considered hierarchically superior to economic rights. Concerning data relating to a specific individual, Article 4.1 1) GDPR defines personal data as anything "relating to an identified or identifiable natural person". Some scholars have warned about the dangers that an overly large definition of personal data might have on the applicability of the raison-d'être of this regulation [12]. In more technical terms, personal data is information that can lead to the identification of an individual.

Data processing. Further, GDPR is relevant for the application of the AI model as its operations can be described as data processing when performed on personal data. Article 4.2 GDPR describes processing as practically any operation involving personal data. Therefore, whenever an AI model uses personal data (to train or to give a specific output autonomously), there will be a processing activity. The GDPR defines certain categories of more sensitive personal data in Article 9.1 GDPR. These include biometric, genetic, and general 'health data' (the latter are better described in Article 4.1 15) GDPR). These could also be categories of data used by the model. Sensitive data in principle should not be processed but some exceptions can justify such operations. In Article 9.2 h) GDPR increasing the efficiency of a medical diagnosis (besides the expressed specific consent of the patient, see 9 2. a)) can be a justification for sensitive personal data processing. In this model, the function of the data processing is not to identify people but to create a technology that averts discriminatory results and can increase the efficiency of medical diagnosis. This model would also fulfill the respect of the principle of fairness as set in Article 5 1. a) GDPR.

The stakeholders: accountability and liability. Two main stakeholders can be involved in the development of AI technologies from a GDPR perspective. The first one is the controller (Article 4.7 GDPR) which is the "natural, legal person or public authority... which, alone or jointly with

others determines the purposes and the means of the processing of personal data". In our particular case, the controller should be either the public or private actor that decides to use the model. However, the subject/entity which handles concretely the data processing phase is called the processor (Article 4.8 GDPR) and could be a company to which these processing functions are delegated by the controller. The GDPR, by stating the principle of accountability (Article 5.2 GDPR) and of responsibility/liability of the controller (Article 24 GDPR and Article 82 GDPR), mandates that the same controller must implement the technical measures that are requested in order to ensure the security (Articles 32-35 GDPR) of the data subject (the person whose data are being processed). This means that the entities responsible for the training and development of neural networks will not be exempt from these rules whenever data processing activities involve personal data. Security and safety of personal data are expected by the controller as a concrete application of the principles of fairness, transparency, and lawfulness (Article 5.1 a) GDPR).

Data protection by default and by design as a tool. The GDPR is a system with a regulatory and compliance function, but, at the same time, it is open-ended as it envisions the protection of personal data as a result and does not focus on how to achieve this target, as it depends also by the technical and economic means available in each specific case. That is why Article 25 of GDPR sets out the principle of privacy by design and by default. It means that privacy should be the objective through which all the processing activities of a data-driven technology must aim from their very first conceptualisation [13] and, also, privacy should be the default option, which means that it should revolve around the amount of data that is necessary for the processing and not to exceed that limit (this principle is called data minimization principle and can be found at Article 5.1 c) GDPR) [14]. The general character of this article also leaves way not only to business and management methods but also to the technology itself in the task to find ways that can increase the privacy and security of the data subject [15]. The model could be a concrete application of this principle as it protects the identity of the subject, hence their privacy, from the first conceptualization to its concrete application.

1.2. New perspectives

The proposed regulation for AI by the European Commission adopts two main views on regulating this technology. On the one hand, the focus is on the accountability principle which makes the regulation less detailed than regulations are in general. The scope of that is to make it a piece of legislation that is technologically neutral as much as possible and to resist the evolution of this kind of technology through time. On the other hand, risk assessment is the rationale through which this regime works. There are three main kinds of AI according to the regulation. Firstly, there are the AI algorithms that are forbidden and consist of technologies that allow the surveillance and the subliminal persuasion of people and especially of minorities and identifiable groups of people. There are specific exceptions that are provided for public authorities' surveillance functions. Secondly, there are the so-called high-risk AI technologies which for the moment are listed in Annex I of the proposal. They include all kinds of machine learning, logic, and knowledge-based approaches (including knowledge representation, inductive and logic programming, inferences and deductive engines, symbolic reasoning, and expert), statistical approaches, Bayesian estimation, and search and optimization methods. These technologies must be evaluated for their objective and the risk they might entail for EU citizens. To this extent, there will be an EU database for the high-risk AI algorithms which will be open and accessible to all EU citizens (Article 60 of the proposal). Each Member state will have to set up an authority (or confer new competencies to old ones) in order to check on the work of specific Notified Bodies (Title III, Chapter 4). These bodies will be private, semi-private, or public organizations with the task of certifying that AI algorithms are state-of-the-art and secure. There will be a European Artificial Intelligence Board (Title VI) which will be composed of the national agencies or authorities which will be in charge of the coordination and monitoring of the algorithms on the market. This proposal, if voted favorably by the European Parliament and the Council of the EU, will become bidding in 27 states and there is some reason to believe that it will serve as a legal model for other countries in the future just as the GDPR did.

1.3. Technological context

Towards privacy preservation, cryptographic techniques cover a prominent role and have broad applications. In particular, attribute-Based Encryption (ABE) is a promising cryptographic primitive which is able to implement

access control for secure data storage, for example, in the cloud [16, 17], or even probabilistic data structures [18]. Besides, homomorphic-based encryption involves the necessity to have a secret key to access the data and is one of the most used approaches to data protection [19]. Classical approaches like the above-mentioned, however, protect the data entirely, without leakage. Nowadays, users are more and more willing to share content to get services, sharing even personal data in order to get a service. Standard cryptographic techniques are inapplicable in these contexts, especially considering that most of the data processing approaches involve the utilization of Artificial Neural Networks (ANNs). Towards this end, the proper design of an algorithm, which is not entirely encrypting the data but which filters a part of the information to be kept private, is necessary.

Privacy-aware learning is not a novel concept in machine learning. One of the very first works in such an area was published in the far 1965 by Warner [20]. In particular, there were suggested privacy-preserving methods for survey sampling. Following this path, in the 70s many works have been proposed in different areas like census taking and analysis of tabular data by Fellegi [21]. Very recently, thanks to the increase in computational capabilities, many works have been proposed on privacy-preserving in computational frameworks. A work by Dwork et al. [22] studied how much noise is required to guarantee "differential privacy" [23] from data. Following these paths, a more recent work by Duchi et al. [24] formalized convergence boundaries for training and the trade-off between privacy guarantees and the utility of the resulting statistical estimators. This knowledge has been also recently applied to deep learning frameworks, with a work by Abadi et al. [25], by introducing some tuned noise in the update rule.

A very recent work by Chamikara et al. [26] proposes the insertion of a "randomizing layer" to guarantee anonymization in the features used for training. While this approach certainly guarantees no locally-sensitive information leak under some constraints, it is a very limiting approach as it requires randomization to happen after convolutional layers. We know nowadays most the ANNs have few fully-connected layers or they are even fully convolutional, and the proposed approach is not applicable without the further insertion of fully-connected layers. Furthermore, it introduces computational overhead, and full back-propagation on the entire model is not suitable as no assumption on the random sampling can be done (so the entire convolutional part must be pre-trained in a non-private fashion and is not further modified).

A different approach to preserving data privacy is the so-called "federated

learning" approach. In general, private datasets are held by the proprietary of the data, who is directly training a local neural network model. Now, the parameters of the models are sent to a master node, which is then propagating to all the private computational nodes the general configuration of the parameters. This approach has been proposed by Shokri and Shmatikov [27] and allows parallel and private computation. However, such an approach does not take into account any ethical bias, like gender or race: what it guarantees is that the original data are not directly shared, but some sensible information is.

In our framework, we train a deep neural network model on some target classes, but we desire at the same time to minimize the risk of information leakage for some classes we desire to keep private. Towards this end, we propose a regularization strategy to Disentangle Private classes: we define a bottleneck layer Γ in our model and we de-correlate features belonging to the same bias class but having a different target. We have two contributions to DisP: a memory term, which keeps track of the average correlation in the full training set, and a local mini-batch term which orthogonalizes all the features belonging to the same bias class but different targets. To the best of our knowledge, DisP is the first approach in this context having generality towards the neural network architecture (or in other words, it applies to any deep learning model), providing insights on the latent space representation of the information, and practically showing the features' correlations. With this work, we move the first steps to "open the black box" ANN models, and we accomplish this by putting a constraint on the information bottleneck of our model, making comparisons with other approaches impossible with the same experimental setup.

The rest of the paper is organized as follows. Sec. 2 provides details about DisP working principles, Sec. 3 presents two possible attacks to retrieve the private information, one unsupervised and another one supervised. Sec. 4 introduces biased-MNIST, a dataset where we have full control of spurious correlations we desire to keep private from the model, providing a theoretical analysis and observing DisP's effectiveness compared to the theory. Then, in Sec. 5 we test the effectiveness of DisP on real datasets, and finally, Sec. 6 concludes.

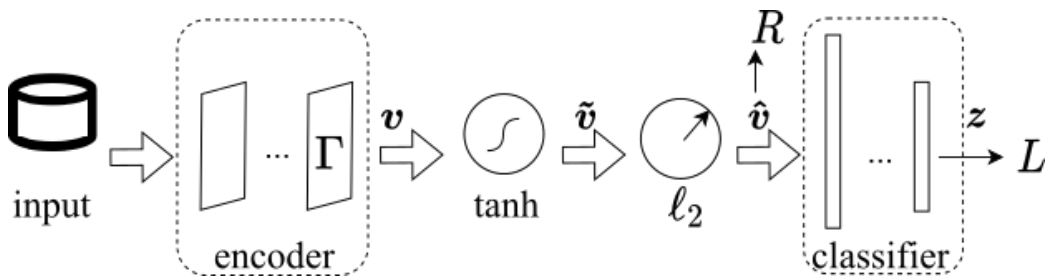


Figure 2: Representation for a generic trained model. The features for DisP are extracted at the output of the encoder (Γ layer), and then tanh activated and normalized to have a unitary norm.

2. Methods and procedures

In this section, after introducing the notation, we present DisP, our proposed regularization term, whose aim is to regularize the deep features aiming at reducing the propagation of the chosen private features onward in the deep model. Towards this end, we are going to support the description of our proposed technique by introducing a toy dataset, biased-MNIST, where we have full control over the impact of some private features over the learning/inference process. Despite such a dataset has been originally designed for debiasing applications, we can consider the biased features as the private ones. Then, we will introduce our DisP term and we will observe its impact during learning.

2.1. Preliminaries

Let us assume we focus our attention on the output of Γ, which is the output layer of the encoder for our model (Fig. 2). We say the output of Γ for the i-th sample is $\boldsymbol{v}_i \in \mathbb{R}^{N_\Gamma \times 1}$. DisP will be a regularization term, computed over features extracted from Γ and undergoing proper activation (tanh and ℓ_2 normalization, as displayed in Fig. 2), to be minimized besides the overall loss of the model, computed at the output of the classifier. The effect we desire to achieve is to have the part of the deep model before Γ to select features that are important to solve the target task, filtering all of those containing private information. The overall objective function we aim to minimize with our learning is the classical formulation

$$J = \eta L + \gamma R_\perp, \tag{1}$$

Table 1: Overview of the notation used in this work.

Symbol	Meaning
M	minibatch size
$M_{t,p}$	cardinality of the samples having the same target t and the same private feature p
C	cardinality of targets
N_Γ	output size of Γ
$\boldsymbol{v}_i$	i-th sample in the minibatch, after Γ
$\hat{\boldsymbol{v}}_i$	i-th sample in the minibatch, after the normalization layer
$\mathcal{T}(\boldsymbol{a}_i)$	function which extracts the target class of $\boldsymbol{a}_i$
$\mathcal{P}(\boldsymbol{a}_i)$	function which extracts the private feature class of $\boldsymbol{a}_i$
δ_{ab}	Kronecker delta
$\bar{\delta}_{ab}$	$1 - \delta_{ab}$
$\langle \boldsymbol{a}, \boldsymbol{b} \rangle$	scalar product between $\boldsymbol{a}$ and $\boldsymbol{b}$ transposed.
$\mathbb{P}(a)$	probability of a
$\mathbb{H}(a)$	entropy of a
$\mathbb{I}(a, b)$	mutual information between a and b

where L is the loss function for the trained task, η and γ are positive hyperparameters, and $R_\perp$ is our proposed DisP. A summary of the notation used through this work is provided in Table 1: as a standard approach, bold values mark vector quantities.

2.2. Bottleneck layer

To address our regularization term, we need first to decide the layer where we wish to apply our DisP regularization, whose task is to prevent information leakage (or in other words, what Γ should be). Due to the complexity of currently-deployed deep neural network architectures with residual layers [28, 29] and skip connections [30], and consider some theoretical frameworks analyzing the information flow in a deep learning model [31, 32], a reasonable choice for our elected information bottleneck layer is the one before the classification layers, or in other words, between the encoder and the classifier.

For our purposes, we desire the extracted features $\boldsymbol{v}$ to have unit norm; hence, we insert in the model a *normalization layer*:

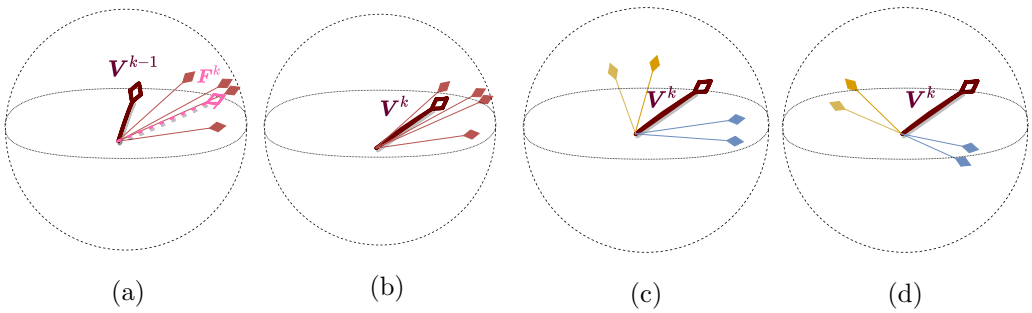


Figure 3: Effect of the $R_{\perp}^{mem}$ term. The same arrow's point represents the same private class p while the same arrow's color indicates the same target t. First, the average $\boldsymbol{F}_{t,p}^{k}$ for the current mini-batch is computed (a), then the memory term $\boldsymbol{V}_{t,p}^{k}$ is provided (b). On the features with the same bias but different target (c) the penalty $R_{\perp}^{mem}$ is applied (d).

- First, it takes the pre-activation $\boldsymbol{v} \in \mathbb{R}^{N_T \times M}$ and a tanh activation function is applied to it:

$$\tilde{v}_{ij} = \tanh(v_{ij}) \tag{2}$$

  After this, $\tilde{v}_{ij} \in (-1; +1)\ \forall i, j$.

- Then, an ℓ_2 normalization is applied to the entire feature vector $\tilde{\boldsymbol{v}}_i$:

$$\hat{\boldsymbol{v}}_i = \frac{\tilde{\boldsymbol{v}}_i}{\|\tilde{\boldsymbol{v}}_i\|_2}. \tag{3}$$

The DisP regularization function will be applied to the normalized $\hat{\boldsymbol{v}}_i$ values. The typical ReLU activation function allows feature values in the range $[0; +\infty)$. For our purposes, we need the feature values where DisP is applied to be limited: we will empirically observe that including this activation at the bottleneck layer does not significantly impact the performance of the model.

2.3. Disentangling the Private information

To disentangle the private features at training time, we are required two terms:

- the *average* features for every target t and every private class p, over all the dataset;

- the *current* features for every target t and every private class p, over the computed minibatch.

What we aim to accomplish here is to disentangle features belonging to the same private class b but different target classes. In this way, differently from other works like [33] and [34], we will minimize the risk of destroying non-private information in the process, the same time preventing private data leakage and maintaining the performance high.
For this reason, we first update the average private features by computing the average features for the current mini-batch:

$$\boldsymbol{F}_{t,p}^{k} = \frac{1}{M_{t,p}} \sum_{i} \hat{\boldsymbol{v}}_i \cdot \delta_{t,\mathcal{T}(\hat{\boldsymbol{v}}_i)} \cdot \delta_{p,\mathcal{P}(\hat{\boldsymbol{v}}_i)} \tag{4}$$

with k as a mini-batch index, and we update the memory term

$$\boldsymbol{V}_{t,p}^{k} = (1-\beta) \cdot \boldsymbol{V}_{t,p}^{k-1} + \beta \cdot \boldsymbol{F}_{t,p}^{k}, \tag{5}$$

where β is the momentum coefficient. Then, we can compute a memory-based disentangling term for the current mini-batch

$$R_{\perp}^{mem} = \frac{1}{M \cdot (T-1)} \sum_{t=1}^{T} \sum_{i=1}^{M} \left| \langle \hat{\boldsymbol{v}}_i, \boldsymbol{V}_{t,p}^{k} \rangle \right| \cdot \bar{\delta}_{t,\mathcal{T}(\hat{\boldsymbol{v}}_i)}. \tag{6}$$

Minimizing (6) equals minimizing the average of the correlations between features belonging to the same private class but different target classes. A visual representation of the computation of this term is depicted in Fig. 3. However, minimizing this term is not a sufficient condition to de-correlate all the features: there can still be formed clusters mapping the private information. For example, looking at Fig. 3d, the features in blue and yellow are de-correlated from $\boldsymbol{V}_{t,p}^{k}$, but the two classes are still anti-correlated. Towards this end, we also have a term that de-correlates *all* the features within the same mini-batch, mapped to the same bias but having different targets:

$$R_{\perp}^{batch} = \frac{1}{M} \sum_{i=1}^{M} \frac{1}{\sum\limits_{t \neq \mathcal{T}(\boldsymbol{v}_i)} M_{t,\mathcal{P}(\hat{\boldsymbol{v}}_i)}} \cdot \sum_{j} \left| \langle \hat{\boldsymbol{v}}_i, \hat{\boldsymbol{v}}_j \rangle \right| \cdot \bar{\delta}_{t,\mathcal{T}(\hat{\boldsymbol{v}}_i)} \tag{7}$$

This term de-correlates all the features belonging to the same private class but different target classes within the mini-batch. A visual effect of $R_{\perp}^{batch}$

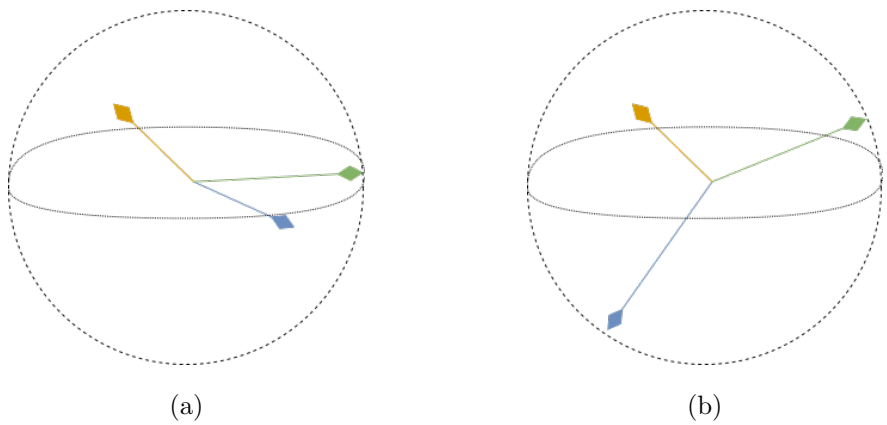


Figure 4: Effect of the $R_{\perp}^{batch}$ term. For each i-th example, the others having the same bias but different targets are selected - in (a), the yellow vector is out i-th. Then, these vectors are each-other de-correlated (b).

is shown in Fig. 4. Having this term alone, with no memory term, for a large dataset, might result in an oscillatory state, where some features might be re-correlated after a certain number of epochs. Towards this end, the memory term in (6), despite modeling only the average correlation among all the private features, helps in a long-range average de-correlation. The two contributions in (6) and (7) are simultaneously minimized, and together they constitute the DisP regularization term:

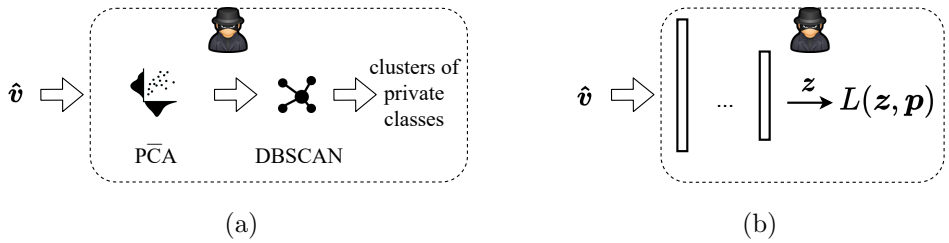
$$R_{\perp} = \gamma^{mem} \cdot R_{\perp}^{mem} + \gamma^{batch} \cdot R_{\perp}^{batch} \tag{8}$$

where γ^{mem} and γ^{batch} are two positive hyper-parameters, which have typically the same value (in such cases, we will refer to those simply as γ).

3. Attacks

To assess potential attacks aiming at retrieving the private features, we assume a potential attacker has access to $\hat{\boldsymbol{v}}_i$.
Here, a typical approach to mining private information is to use an unsupervised approach. Toward this end, a possible attack involves the following steps:

- Compute the PCA over the set of available data. Through this, it is possible to reduce the dimensionality of the data. A typical good choice is to extract the components such that the energy content is at least



(a) (b)

Figure 5: Possible class inference attacks on the regularized $\hat{\boldsymbol{v}}$: unsupervised attack (a) and worst-case supervised attack (b).

95% over the total energy. We assume a worst-case scenario where the attacker has *all* the possibly available data (hence, the whole of the training/test sets);

- Run DBSCAN to find data dependencies. Despite an attacker might know $\hat{\boldsymbol{v}}_i$ is normalized over a unitary hyper-sphere and might be tempted in using special clustering algorithms designed for this scenario like spherical k-means [35], the typical high-dimensionality of the data makes more useful to apply first a PCA to reduce such dimensionality. After the PCA is applied, the spherical constraint typically does not hold anymore; hence, a more general approach, like DBSCAN, which can capture non-trivial data dependencies, should be deployed.

Once clusters of data are found, we can look at the content of each of these clusters: if all of these contain a homogeneous number of data points belonging to the same private class, then the private information has not been extracted, and the attacker failed. On the contrary, if there are clusters containing a high percentage of data points with the same private class, then the attack has success and the private information has been retrieved. We call this *unsupervised attack*.

As an upper-bound case, let us assume the attacker has the training set data provided with the private class label, which is hence known. In such a case, the attacker is willing to train a classifier from $\tilde{y}_i$, attempting to learn the relationship between the input $\tilde{y}_i$ and the private features b. Such an attack has success if the trained model, which in our case will be a multi-layer perceptron (MLP), can generalize well on the test set data. If the information about the private classes has been erased, the MLP will be forced in a *memorization* state, where the training set will necessarily overfit over the training set, if the number of parameters in it is sufficiently large. We call this *su-*

Figure 6: Some samples from biased-MNIST [36]: the background colors (our private feature) highly correlate with the digit classes (the target).

pervised attack which constitutes an upper bound to the learnable private information given our testing methods. A graphical summary of the attacks is provided in Fig. 5.

4. A controlled experiment

In this section we are going to conduct experiments over biased-MNIST, a dataset where we have direct access and control to parameters directly tuning correlations between input features and some extra features not directly correlated to the target task we desire to train a model on. This allows us to address a simplified theoretical framework that gives us insights into the effect of DisP over the information extracted from the available data.

4.1. The biased-MNIST dataset

This dataset has been recently proposed by Bahng *et al.* [36] for debiasing purposes; however, we can consider the biased features of such a dataset as the features we desire to hide from the classifier. This dataset is constructed from MNIST [37] by injecting color into the background of the images, as shown in Fig. 6. In this dataset, we have ten different targets (hence, $C = 10$). Each digit is associated with one of ten pre-defined colors, which will be our private feature. To assign the background color to an image of a given target class, the pre-defined color is selected with a probability ρ, and any other color is chosen with a uniform probability $(1 - \rho)$:

$$\begin{cases} \mathbb{P}(\mathcal{T}(\boldsymbol{x}) = i | \mathcal{P}(\boldsymbol{x}) = i) = \rho \\ \mathbb{P}(\mathcal{T}(\boldsymbol{x}) \neq i | \mathcal{P}(\boldsymbol{x}) = i) = \frac{1}{9}(1 - \rho). \end{cases} \tag{9}$$

Straightforwardly we can assume

$$\mathbb{H}(T) = \mathbb{P}(t) \log_{10}(0.1) = 1 \tag{10}$$

$$\mathbb{H}(P) = \mathbb{P}(p) \log_{10}(0.1) = 1, \tag{11}$$

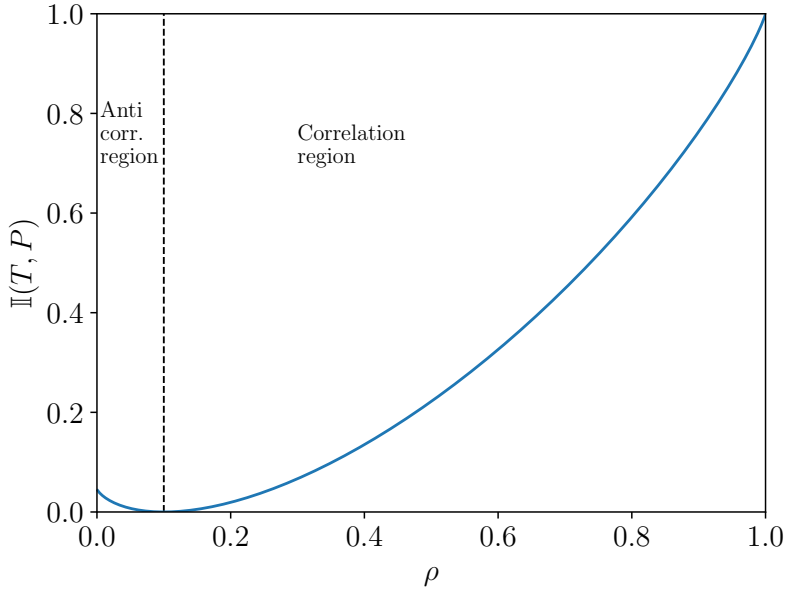


Figure 7: Plot of the mutual information between T and P. We have decorrelation for $\rho = 0.1$.

where T and P are the random variables associated with the target and the private class, respectively. From this, can write the conditional entropy

$$\mathbb{H}(P|T) = \rho \cdot \log_{10}\left[\frac{1-\rho}{9\cdot\rho}\right] + \log_{10}\left[\frac{9}{1-\rho}\right]. \tag{12}$$

Here, we can write the mutual information

$$\mathbb{I}(P,T) = 1 - \rho \cdot \log_{10}\left[\frac{1-\rho}{9\cdot\rho}\right] + \log_{10}\left[\frac{9}{1-\rho}\right]. \tag{13}$$

Figure 7 shows the mutual information between the target X and the private feature Y. Of course, the more the target correlates to the private feature, the highest the chances of having some information leakage. In the next we are going to quantitatively analyze the information leakage on a trained model, introducing the random variable Z which models the output of the model.

4.2. Expected information leakage in biased-MNIST

Let us assume here a real case where the trained model is not a perfect learner, meaning that

$$\mathbb{H}(Z|T) \neq 0 \tag{14}$$

where Z is the random variable associated with the output $\boldsymbol{z}$. The model does not correctly classify the target for two reasons.

- It gets confused by the private features, and it tends to learn to classify on top of those. We model the tendency of learning private features as b: the lower this parameter, the more the model learns the features we desire to keep private, introducing errors in the model.
- Some extra error, unrelated to the bias features, which can be due to stochastic unbiased effects, underfit, or other high-order dependencies between data. We are not strictly interested to model such an effect for our purposes, and we consider this extra error always be zero.

We can now write the joint probability:

$$\mathbb{P}(t,p,z) = \frac{1}{10} \cdot \left[\delta_{tpz}\rho + (1-\delta_{tz})\delta_{tp}\frac{(1-b)(1-\rho)}{9} + (1-\delta_{tp})\delta_{tz}\frac{b(1-\rho)}{9} \right] \tag{15}$$

where $b \in [0;1]$. From this, we can compute the marginal probability

$$\mathbb{P}(p,z) = \frac{1}{10} \cdot \left\{ \delta_{pz} \left[\rho + \frac{1-\rho}{9} \sum_m b(1-\delta_{pm}) \right] + (1-\delta_{pz}) \left[(1-b)\frac{1-\rho}{9} \right] \right\}. \tag{16}$$

From this, we can plot the mutual information between the neural network prediction and the source (Figure 8).

We observe that for high ρ values the mutual information between the information the model learns and the private classes is high, independently from how much the model learns to extract from the private features (b). Our objective here is to discourage the leakage of private information on new data: if we keep the value of $b \approx 0.9$, the model still can keep good performance, but when tested on un-correlated data ($\rho = 0.1$), the mutual information drops. On the contrary, for low values of b, $\mathbb{I}(Z,P)$ remains high, meaning that there is private information leakage.

4.3. Training with DisP

We use the network architecture proposed by Bahng *et al.* [36], consisting of four convolutional layers with 7×7 kernels. As a bottleneck layer here we choose the average pooling layer, before the fully connected classifier of the network. We optimize the model using stochastic gradient descent with

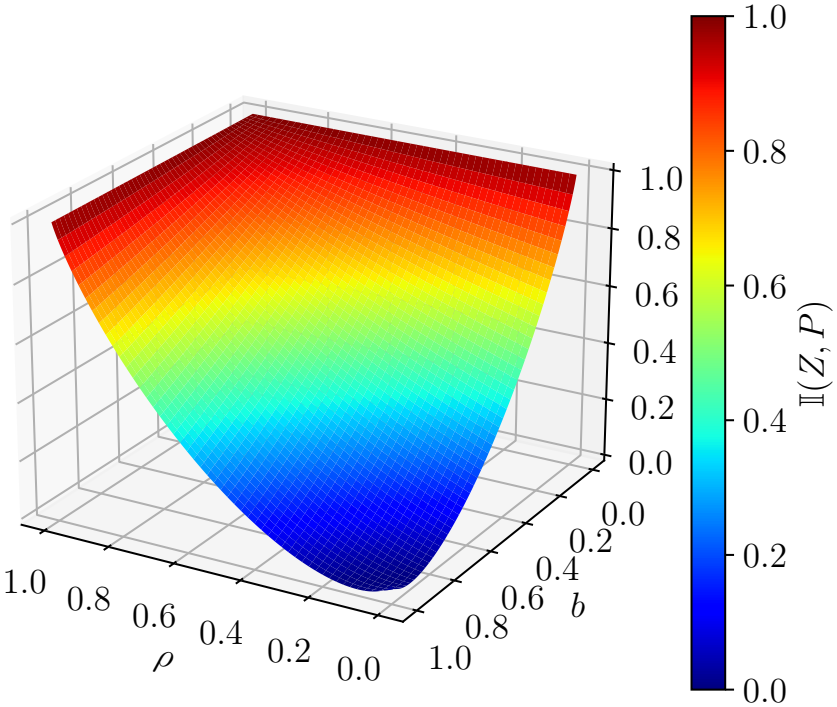


Figure 8: Plot of the mutual information between the private feature and the output of the model. In this case, all the b are equal for visualization purposes.

a learning rate of 0.1, batch size 100, and weight decay 1e-4, for 50 epochs. Table 2 shows the experimental results achieved by training 5 models per configuration. In this case, we are also able to provide an empirical estimation on b (which is the tendency of the model to learn the private features, as discussed in Sec. 4.2).

In the case where there is no correlation between the target and private feature ($\rho = 0.1$), and the tendency b is low, also the information relative to the private feature correlation (R) is typically low, and further minimizing it is not harming or even improving the generalization capability of the model. However, when the correlation between the target and private feature is extremely high ($\rho = 0.99$), the tendency value b drops, meaning that the model is learning the private information in place of the target. This is due to the very high correlation between the target and private feature, which is ρ by definition. In such a case, including the regularization term in the cost

Table 2: Experiments on biased-MNIST. (*) values rounded to the closest decimal.

ρ	γ	b	$R_\perp$	Accuracy [%] ($\rho = 0.1$)
0.1	0	0.9*	0.01	98.34
	0.1	0.9*	1e-3	98.21
	0.2	0.9*	3e-5	98.03
0.99	0	0.21	0.44	89.10
	0.1	0.75	0.12	95.01
	0.2	0.83	0.07	93.73

function is minimizing the correlation term R and also beneficially impacts the accuracy of the model itself. Indeed, the tendency b is increasing, meaning that the model is no longer learning the private feature but is looking at the correct features to extract.

4.4. Attacks

Considering that in this case, we have a theoretical framework estimating the upper bound over the private feature which can be extracted, we will here conduct the unsupervised attack only. We run attacks here for $\rho = 0.99$, with $\gamma = 0$ (which is our baseline model) and $\gamma = 0.1$. We run the attacks over the models showing the median accuracy over the test set. Figure 9 qualitatively shows the outcome of the clustering algorithm. In particular, the different shape represents the cluster found by the PCA+DBSCAN approach, the internal color is the target while the border color is the private features. Even qualitatively, we observe that without applying DisP, there is a significant part of the clusters which map to the private feature, while applying DisP this percentage drops dramatically, averagely to 14%.

4.5. Ablation study

Here we propose an ablation study on DisP. In particular, we evaluate the impact of the two regularization contributes, the memory term $R_\perp^{mem}$ (6), and the mini-batch term $R_\perp^{batch}$ (7). We evaluate this in the challenging scenario where $\rho = 0.99$. Table 3 shows the ablation study results obtained: even though in some cases the weight in the regularization loss (γ) is zero, we still measure and report it. While applying singularly the two contributions are

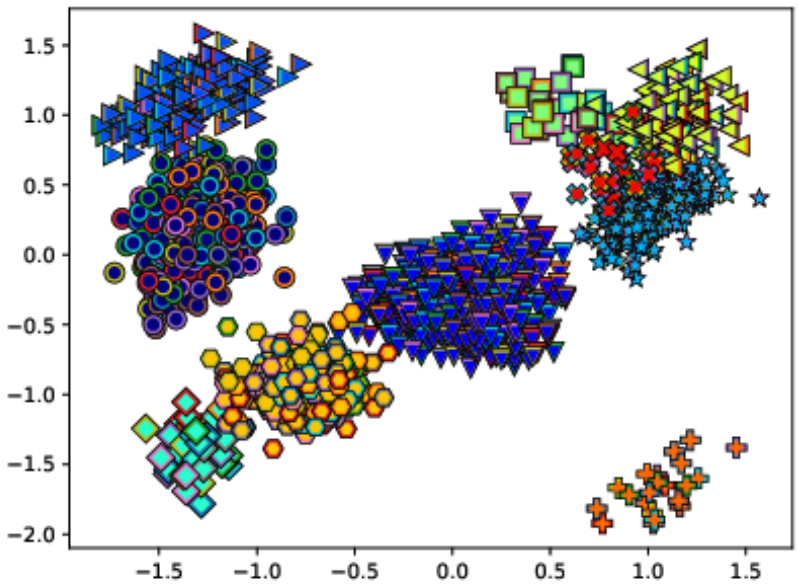


Figure 9: PCA+DBSCAN output for a model regularized with DisP and trained on biased-MNIST, with $\rho = 0.99$ and $\gamma = 0.1$, on the test set. Here the first two principal components are displayed. A different symbol is a different cluster found by DBSCAN, internal color is the target class while the edge color is the private class.

indeed minimizing the overall R in both cases, the lowest value for R, and in this case, the highest accuracy is achieved by applying the same regularization contribution to both. Of the two terms, the largest contribution is overall provided by $R_{\perp}^{batch}$, which disentangles all the feature vectors in the name minibatch.

5. Experiments on real datasets

In this section, we experiment DisP's effectiveness over two real datasets. Our goal here is to train the ANN model on the target classification task applying our regularization strategy over some features we wish to keep private (in our case, to maintain consistency over the experiments, we want to keep private the information over gender). We will show that, in our context, such information is detectable in more trivial cases (like in face image datasets)

Table 3: Ablation study on biased-MNIST ($\rho = 0.99$).

γ^{mem}	γ^{batch}	$R_{\perp}^{mem}$	$R_{\perp}^{batch}$	$R_{\perp}$	Accuracy [%] ($\rho = 0.1$)
0.1	0	0.04	0.18	0.22	91.23
0	0.1	0.07	0.09	0.16	92.54

and in less straightforward cases (like in chest X-ray radiographic images). Our training and inference algorithms are implemented in Python, using PyTorch 1.12 and an RTX2080 Ti NVIDIA GPU with 11GB of memory has been used for training and inference.[2] All the used hyper-parameters have been tuned using grid search.
The results will be presented for two different datasets: Celeb-A and SIIM-FISABIO-RSNA COVID-19 (SFR). After training has been completed, the attacks proposed in Sec. 3 will evidence the possibility of extracting the private information from the bottleneck features $\hat{\boldsymbol{y}}_i$. As a chosen architecture, we select the model which has achieved the lowest validation loss along the training.

5.1. Datasets and setup

CelebA. This dataset [38] has been designed for face-recognition tasks, providing 40 attributes for every image. The dataset contains a total of 202.6k images and, following the official train-validation split, we obtain 162.7k images for the training set, 19.9k images for the validation set, and 19.9k images for testing our models. For our training purposes, we use a ResNet-18 model, choosing the adaptive pooling layer as the bottleneck layer Γ. The training has been performed using SGD, with an initial learning rate of 0.1, decayed by a factor of 10 after no improvement over the validation set loss has been detected for 10 consecutive epochs. The training stops when the learning rate drops below 1e-3. We use batch size 100 with a momentum of 0.9 and weight decay of 1e-5. Images are here re-scaled to 224×224 resolution.
We select here as classification target the recognition of the *Eyeglasses*, *Blond-Hair*, and *Heavymakeup* attributes, while the gender is the private class.

[2]The source code will be made available upon acceptance of the article.

While the eyeglasses ($\rho = 0.62$) and blonde hair ($\rho = 0.58$) attributes are relatively easy to disentangle from the gender, disentangling the heavy makeup attribute is a more challenging choice ($\rho = 0.82$), as there is a high correlation between these attributes (most of the women in the dataset have a heavy makeup).

SIIM-FISABIO-RSNA COVID-19. The currently available dataset[3] comprises more than 6k chest X-ray (CXR) scans in DICOM format, anonymized according to the current GDPR guidelines. For study purposes, however, the metadata associated with these scans comprises information about the gender, which will be used as our private class. We split the dataset into a training set comprising 5k scans, a validation set of 300 scans, and a test set of 600 scans. We train a DenseNet-121 model to classify over the "Negative for Pneumonia" and "Typical Appearance" classes. As a bottleneck layer here we choose, similarly to what was chosen for CelebA, the adaptive pooling layer. The training has been performed using SGD, with an initial learning rate of 0.1, decayed by a factor of 10 after no improvement over the validation set loss has been detected for 5 consecutive epochs. The training stops when the learning rate drops below 1e-3. We use batch size 16 with a momentum of 0.9 and weight decay of 1e-4. The scans are converted using the meta-information contained in the DICOM files and re-scaled to 448×448 resolution.

5.2. Results

Table 4 shows the achieved results. We report the performance on the training set, the accuracy in retrieving the private features with the unsupervised attack, and the train/test accuracy when training an MLP model with one hidden layer of size 300 (1H) or two hidden layers of size 600-300 (2H). Looking at the pure performance at training time (or in other words, the performance for the given target) we observe that in most cases, the use of DisP is not significantly harming the performance. Differently from what was observed in biased-MNIST in Sec. 4.3, there is overall no improvement in the performance when DisP is applied either. This means that the gender information is not deteriorating the performance, but on the contrary, it can be useful for higher generalization. Indeed, we observe a slight deterioration

[3] `https://www.kaggle.com/c/siim-covid19-detection/data`

Table 4: Results on the trained models with the original train-validation-test split.

Dataset	**Target**	**Training**			**Attacks**		
		γ	R_{perp}	Accuracy [%]	Unsup. [%]	Sup.(1H) [Train%-Test%]	Sup.(2H) [Train%-Test%]
CelebA	Blondhair	0	0.38	95.68	73.1	85.8-79.3	91.2-84.8
		0.1	0.04	95.88	57.3	86.3-64.8	93.0-58.4
		0.2	0.02	95.32	55.5	83.7-61.4	90.3-55.8
	Eyeglasses	0	0.08	99.65	51.3	81.3-52.3	88.9-51.2
		0.1	2e-3	99.60	53.2	83.5-53.5	89.1-52.3
		0.2	3e-4	99.60	52.6	82.1-51.7	88.3-51.9
	Heavymakeup	0	0.39	90.30	84.1	94.6-94.5	94.6-94.6
		0.1	0.13	90.33	83.3	94.1-94.1	94.5-94.5
		0.2	0.03	89.79	75.1	93.6-93.3	93.7-93.0
SFR	Pneumonia/Typical	0	0.47	78.12	75.2	83.2-82.4	85.7-81.9
		0.1	0.12	78.43	53.6	67.9-58.4	69.5-60.6
		0.2	0.08	78.02	52.9	65.6-58.9	73.7-57.8

of the performance for the blond hair and heavy makeup cases. Indeed, this can be observed by the higher difficulty in minimizing the R_{perp} term besides the loss, indicating that the information we desire to keep private is not only learned but actively used for the classification task.

Interestingly, we observe a typical trend in all of our experiments: for the baseline models (or in other words, when $\gamma = 0$), the maximum value for the recorded value of R, which results also in the maximum private information leakage, is recorded nearby the lowest value of the loss for the validation set. An example of this typical trend we observe in all the conducted experiments is shown in Fig. 10, which shows the trend of R, the loss on the validation set, and the loss on the test set for the SFR dataset. Intuitively, we could infer that the information we wish to keep private is necessary to generalize on the given task, and as the model starts over-fitting such information is being obfuscated. However, looking at the performance recorded when applying DisP, for $\gamma \neq 0$, we observe that good generalization performance for low values of R is still possible. This means that in this case there is some natural tendency for the model to implicitly learn this private information, which can be discouraged by applying DisP.

For completeness, in Tab. 5 we also propose the same set of experiments with a 5-fold cross-validation strategy, obtaining similar results as those achieved with the original validation set proposed in CelebA and SFR.

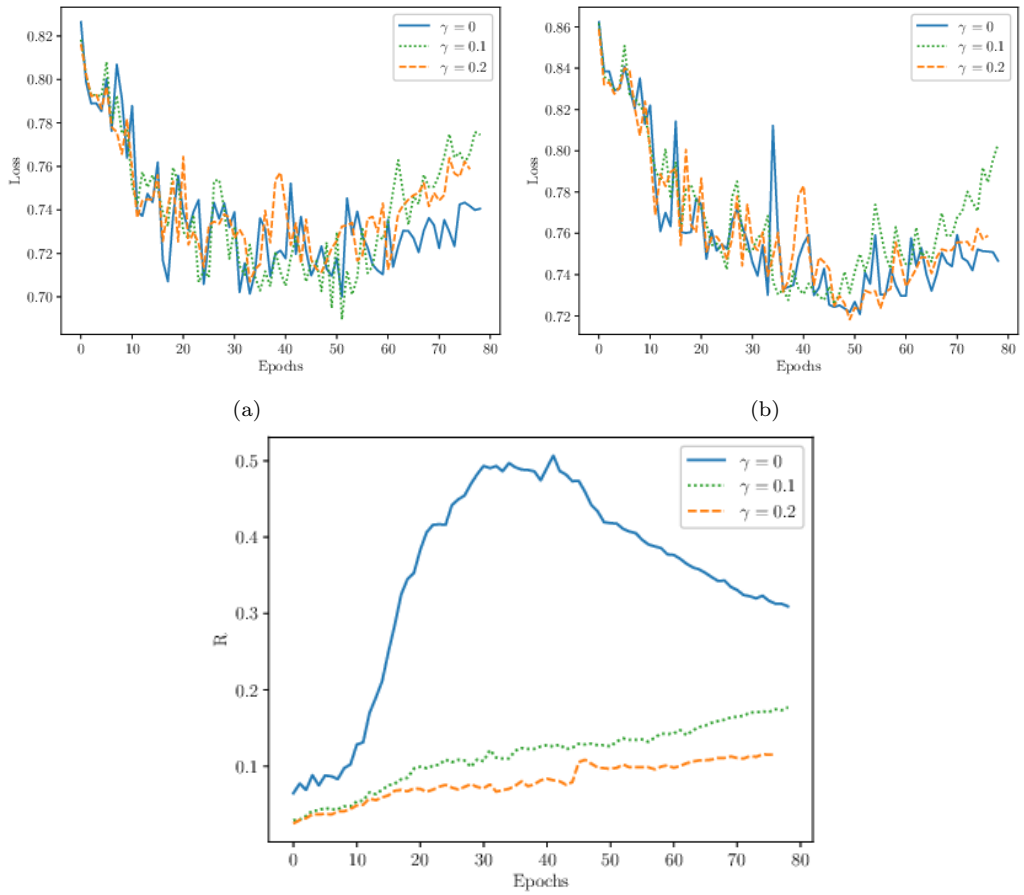


Figure 10: Training on SFR: validation set loss (a), test set loss (b) and R value (c).

5.3. Attacks

Looking at the unsupervised attacks, we interestingly observe that there are certain scenarios in which the information is easier to extract. For example, when the target is Blondhair in CelebA, the unsupervised attack can recover 73% of the private information, evidencing a problem of information leakage. Such a problem is even more evident with the SFR dataset, where an unsupervised attack recovers the private features with 75% accuracy. Of course, there are scenarios in which the private feature is not learned, and, for instance, it is harder to extract: it is the case for the Eyeglasses feature, in the CelebA dataset. Intuitively, such a feature is naturally de-correlated

Table 5: Results on the trained models with 5-fold cross-validation (in the upper row train set performance, in the bottom test set performance).

Dataset	**Target**	**Training**		**Attacks**		
		γ	Accuracy [%]	Unsup. [Train%-Test%]	Sup.(1H) [Train%-Test%]	Sup.(2H) [Train%-Test%]
CelebA	Blondhair	0	95.1 ± 0.1	62.0 ± 0.6	89.8 ± 0.4	90.9 ± 0.3
					88.1 ± 0.6	88.4 ± 0.3
		0.1	95.0 ± 0.1	58.0 ± 0.1	81.2 ± 0.3	81.8 ± 0.3
					80.5 ± 1.0	81.5 ± 0.6
		0.2	95.1 ± 0.2	58.0 ± 0.0	81.3 ± 1.5	81.8 ± 1.5
					80.6 ± 1.7	80.7 ± 1.2
	Eyeglasses	0	99.6 ± 0.0	54.4± 1.3	87.8 ± 0.3	90.2 ± 0.2
					85.2 ± 0.3	85.2 ± 0.5
		0.1	99.7 ± 0.0	51.2± 0.3	79.0 ± 0.8	79.3 ± 0.9
					78.6 ± 1.1	78.7 ± 1.1
		0.2	99.7 ± 0.0	50.7± 1.2	78.3 ± 0.5	79.0 ± 0.5
					77.6 ± 0.7	78.0 ± 0.5
	Heavymakeup	0	90.3 ± 0.0	86.4± 3.8	94.7 ± 0.0	94.7 ± 0.1
					94.5 ± 0.1	94.4 ± 0.1
		0.1	90.1 ± 0.1	83.2± 0.2	94.5 ± 0.2	94.5 ± 0.2
					94.5 ± 0.2	94.5 ± 0.2
		0.2	90.1 ± 0.2	80.4± 5.7	94.7 ± 0.0	94.7 ± 0.2
					94.6 ± 0.1	94.4 ± 0.2
SFR	Pneumonia/Typical	0	77.1± 1.0	77.5± 3.7	85.2 ± 0.2	86.2 ± 0.2
					82.3 ± 0.1	84.9 ± 0.2
		0.1	78.2± 1.3	57.2± 0.1	70.4± 2.1	70.4± 2.4
					62.6± 1.8	60.5± 1.1
		0.2	77.5± 0.8	58.0± 0.1	67.5± 0.5	69.1± 3.5
					60.0± 1.5	58.7± 1.4

with gender, and for instance, such a piece of information is unlikely to leak from the bottleneck layer.

Looking at the supervised attack, we observe that the performance is typically higher than the unsupervised attack, as expected. While for the CelebA-Eyeglasses scenario, the classifier is (almost) random guessing for all our experiments, that is not the case for the other scenarios. Focusing on the baseline models ($\gamma = 0$), we observe that the generalization performance on the test set shows a tendency to improve by adding more complexity to the classifier (comparing H1 to H2). However, using DisP, we observe a decreasing generalization trend when increasing the classifier complexity. We have

Table 6: Comparison of DisP with other debiasing approaches for the CelebA dataset.

Target	**Approach**	**Accuracy** [%]	**Attacks** Unsup. [Train%-Test%]	Sup.(1H) [Train%-Test%]	Sup.(2H) [Train%-Test%]
Blondhair	Baseline	95.68	73.1	85.8-79.3	91.2-84.8
	ReBias [2]	94.56	87.0	89.8–90.2	90.4–90.9
	RUBi [3]	83.02	68.5	91.0–91.2	91.5–91.7
	Learned-Mixin [4]	50.95	61.6	89.4–89.7	90.0–90.4
	DisP ($\gamma = 0.2$)	95.32	55.5	83.7–61.4	90.3–55.8
Eyeglasses	Baseline	99.65	51.3	81.3-52.3	88.9-51.2
	ReBias [2]	99.64	80.8	90.0–90.1	90.5–90.8
	RUBi [3]	99.70	76.0	89.5–89.6	90.0–90.2
	Learned-Mixin [4]	63.78	61.4	92.4–92.2	92.9–92.9
	DisP ($\gamma = 0.2$)	99.60	52.6	82.1–51.7	88.3–51.9
Heavymakeup	Baseline	90.30	84.1	94.6-94.5	94.6-94.6
	ReBias [2]	90.45	84.0	95.0–95.3	94.9–95.5
	RUBi [3]	86.71	75.4	94.3–94.5	94.4–94.7
	Learned-Mixin [4]	54.93	90.4	94.3–94.5	94.5–94.7
	DisP ($\gamma = 0.2$)	89.79	75.1	93.6–93.3	93.7–93.0

not used any validation/fine-tuning strategy to train the supervised attack: we have used SGD with a fixed learning rate of 0.1, batch size of 100, and weight decay of 0.9 for 50 epochs, for all the experiments. Under these conditions, since the training accuracy increases but the test accuracy decreases, we can say the classifier is driven in a memorization state: this means that it is simply unable to extract general features to learn the private classes, as they are disentangled. As the value of γ increases (meaning that the weight of our regularization term in the learning process increases), the performance achieved by the attacks decreases.

5.4. A comparison to debiasing approaches

Very recent work suggested a parallel between privacy preservation and debiasing [1]. More specifically, this work suggests the possibility of employing debiasing approaches for privacy preservation, still preserving the original setup for which the debiasing approaches are conceptualized. We hereby propose a comparison, in terms of attacks as already presented in Table 4, for the CelebA dataset and our specific setup (balanced classes),

of DisP and three popular debiasing approaches: ReBias [2], RUBi [3] and Learned-Mixin [4].[4] Table 6 reports the results, from which we observe that the proposed approaches tend to amplify the information related to the private classes. This is caused by our specific setup, where the training data are balanced at the source, which makes it difficult to these approaches to properly extract, and consequently weight, the information to keep private. This leads to some potential fails, like for Learned-Mixin, where the private information is the one learned in place of the target one. Other approaches, like ReBias, can achieve close to state-of-the-art performance (beating the baseline for Heavymakeup), but at the cost of a massive utilization of private information. DisP proves to be superior to the compared debiasing approaches in removing private information.

5.5. Applicability beyond deep learning

DisP proposes itself as a regularization strategy towards a disentanglement of private classes, and as such, it is implicitly bound to gradient-based optimization techniques. However, its applicability is not necessarily limited to deep learning models: other machine learning algorithms are compatible with DisP, like for example Support Vector Machines (SVMs), where the impact of DisP will be on the class' separator. Besides, DisP can virtually be applied isolatedly as a perceptron besides any other machine learning strategy (like for example, in a compression pipeline after PCA).

6. Conclusion

In this work, we have proposed DisP, a regularization term that disentangles private classes. In particular, we favor the selection of features correlating with the target class filtering the propagation of features correlating with the private classes. To be more interdisciplinary and to create a more robust and ethical AI technology, the private features selection was made by keeping some key concepts of the GDPR in mind. The GDPR is an important legislative act whose objective is to protect people's privacy not only through regulatory and compliance functions but also through principles

[4]To run these experiments, we use the implementation provided in `https://github.com/EIDOSLAB/bridging-debiasing-privacy-deep-learning` with the default approach-specific hyper-parameters.

such as the one of privacy by default and by design, which is bound to influence emerging technologies, AI included. Once the features to be considered "private" are detected, DisP can disentangle them at some bottleneck layer, minimizing the risk of extraction for this information. We have shown the effectiveness of the models trained with DisP attempting both unsupervised and supervised attacks to retrieve the information. In particular, we have tested on face images and CXR images, showing that in certain cases there is unwanted information leakage it can be successfully recovered unless the models are not DisP regularized. Future work includes the development of a layer able to handle quantized features on top of which mutual information can be directly measured and minimized.

References

[1] C. A. Barbano, E. Tartaglione, M. Grangetto, Bridging the gap between debiasing and privacy for deep learning, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, 2021, pp. 3806–3815.

[2] H. Bahng, S. Chun, S. Yun, J. Choo, S. J. Oh, Learning de-biased representations with biased representations, in: International Conference on Machine Learning, PMLR, 2020, pp. 528–539.

[3] R. Cadene, C. Dancette, M. Cord, D. Parikh, et al., Rubi: Reducing unimodal biases for visual question answering, Advances in neural information processing systems 32 (2019).

[4] C. Clark, M. Yatskar, L. Zettlemoyer, Don't take the easy way out: Ensemble based methods for avoiding known dataset biases, in: K. Inui, J. Jiang, V. Ng, X. W. 0001 (Eds.), Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019, Association for Computational Linguistics, 2019, pp. 4067–4080. doi:10.18653/v1/D19-1418.
URL https://doi.org/10.18653/v1/D19-1418

[5] S. Sonoda, N. Murata, Neural network with unbounded activation functions is universal approximator, Applied and Computational Harmonic Analysis 43 (2) (2017) 233–268.

[6] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-cam: Visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 618–626.

[7] S. A. Osia, A. S. Shamsabadi, S. Sajadmanesh, A. Taheri, K. Katevas, H. Rabiee, N. Lane, H. Haddadi, A hybrid deep learning architecture for privacy-preserving mobile analytics, IEEE Internet of Things Journal 7 (2020) 4505–4518.

[8] 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) 1–88.

[9] Ethics guidelines for trustworthy ai., `https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai`.

[10] Assessment list for trustworthy artificial intelligence (altai) for self-assessment., `https://digital-strategy.ec.europa.eu/en/library/assessment-list-trustworthy-artificial-intelligence-altai-self-assessment`.

[11] Proposal for a regulation of the european parliament and of the council laying down harmonised rules on artificial intelligence (artificial intelligence act) and amending certain union legislative acts com/2021/206 final.

[12] N. Purtova, The law of everything. broad concept of personal data and future of eu data protection law, Law, Innovation and Technology 10 (1) (2018) 40–81.

[13] R. Leenes, S. De Conca, Artificial intelligence and privacy—ai enters the house through the cloud, in: Research handbook on the law of artificial intelligence, Edward Elgar Publishing, 2018.

[14] P. Voigt, A. Von dem Bussche, The eu general data protection regulation (gdpr), A Practical Guide, 1st Ed., Cham: Springer International Publishing 10 (2017) 3152676.

[15] L. A. Bygrave, Data protection by design and by default: deciphering the eu's legislative requirements, Oslo Law Review 4 (02) (2017) 105–120.

[16] M. Horváth, Attribute-based encryption optimized for cloud computing, in: International Conference on Current Trends in Theory and Practice of Informatics, Springer, 2015, pp. 566–577.

[17] Y. Zhang, R. H. Deng, S. Xu, J. Sun, Q. Li, D. Zheng, Attribute-based encryption for cloud computing access control: A survey, ACM Computing Surveys (CSUR) 53 (4) (2020) 1–41.

[18] A. Singh, S. Garg, R. Kaur, S. Batra, N. Kumar, A. Y. Zomaya, Probabilistic data structures for big data analytics: A comprehensive review, Knowledge-Based Systems 188 (2020) 104987.

[19] A. Acar, H. Aksu, A. S. Uluagac, M. Conti, A survey on homomorphic encryption schemes: Theory and implementation, ACM Computing Surveys (Csur) 51 (4) (2018) 1–35.

[20] S. L. Warner, Randomized response: A survey technique for eliminating evasive answer bias, Journal of the American Statistical Association 60 (309) (1965) 63–69.

[21] I. P. Fellegi, On the question of statistical confidentiality, Journal of the American Statistical Association 67 (337) (1972) 7–18.

[22] C. Dwork, F. McSherry, K. Nissim, A. Smith, Calibrating noise to sensitivity in private data analysis, Journal of Privacy and Confidentiality 7 (3) (2016) 17–51.

[23] C. Dwork, J. Lei, Differential privacy and robust statistics, in: Proceedings of the forty-first annual ACM symposium on Theory of computing, 2009, pp. 371–380.

[24] J. C. Duchi, M. I. Jordan, M. J. Wainwright, Privacy aware learning, Journal of the ACM (JACM) 61 (6) (2014) 1–57.

[25] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, L. Zhang, Deep learning with differential privacy, in: Proceedings of the 2016 ACM SIGSAC conference on computer and communications security, 2016, pp. 308–318.

[26] M. A. P. Chamikara, P. Bertók, D. Liu, S. Camtepe, I. Khalil, An efficient and scalable privacy preserving algorithm for big data and data streams, Computers & Security 87 (2019) 101570.

[27] R. Shokri, V. Shmatikov, Privacy-preserving deep learning, in: Proceedings of the 22nd ACM SIGSAC conference on computer and communications security, 2015, pp. 1310–1321.

[28] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016) 770–778.

[29] G. Huang, Z. Liu, K. Q. Weinberger, Densely connected convolutional networks, 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017) 2261–2269.

[30] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: MICCAI, 2015.

[31] N. Tishby, N. Zaslavsky, Deep learning and the information bottleneck principle, 2015 IEEE Information Theory Workshop (ITW) (2015) 1–5.

[32] Z. Goldfeld, Y. Polyanskiy, The information bottleneck problem and its applications in machine learning, IEEE Journal on Selected Areas in Information Theory 1 (2020) 19–38.

[33] E. Tartaglione, C. A. Barbano, M. Grangetto, End: Entangling and disentangling deep representations for bias correction, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 13508–13517.

[34] E. Tartaglione, M. Grangetto, A non-discriminatory approach to ethical deep learning, 2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom) (2020) 943–950.

[35] C. Buchta, M. Kober, I. Feinerer, K. Hornik, Spherical k-means clustering, Journal of statistical software 50 (10) (2012) 1–22.

[36] H. Bahng, S. Chun, S. Yun, J. Choo, S. J. Oh, Learning de-biased representations with biased representations, in: International Conference on Machine Learning (ICML), 2020.

[37] Y. LeCun, C. Cortes, C. Burges, Mnist handwritten digit database, ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist 2 (2010).

[38] Z. Liu, P. Luo, X. Wang, X. Tang, Deep learning face attributes in the wild, in: Proceedings of International Conference on Computer Vision (ICCV), 2015.