EXtremely PRIvate supervised Learning

Armand Lacombe, Saumya Jetley, Michèle Sebag

▶ **To cite this version:**

Armand Lacombe, Saumya Jetley, Michèle Sebag. EXtremely PRIvate supervised Learning. Conférence d'APprentssage - CAP, Jun 2021, St-Etienne, France. hal-03620873

HAL Id: hal-03620873
https://inria.hal.science/hal-03620873v1

Submitted on 27 Mar 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

EXtremely PRIvate supervised Learning

Armand Lacombe *[1], Saumya Jetley*[1], et Michèle Sebag[1]

[1]AO/TAU, CNRS - INRIA - LISN - Univ. Paris-Saclay, France

Abstract

This paper presents a new approach called ExPriL for learning from extremely private data. Iteratively, the learner supplies a candidate hypothesis and the data curator only releases the marginals of the error incurred by the hypothesis on the privately-held target data. Using the marginals as supervisory signal, the goal is to learn a hypothesis that fits this target data as best as possible. The privacy of the mechanism is provably enforced, assuming that the overall number of iterations is known in advance.

Keywords: Privacy, Synthetic data generation, Supervised learning

1 Introduction

In quite a few sensitive domains, such as hospitals, or financial markets, the data curator has access to a large repository of private data, but is unwilling/unable to divulge any of this data, referred to as the *target data*. In-situ analysis is compromised owing to constraints on computational resources or on availability of in-house experts. The novelty of the proposed approach, called ExPriL, is to only require some access to the marginals of the target data, in order to learn a fitting hypothesis, and through it a privacy-preserving synthetic version of the target data. Several approaches, aimed to learn from (very) limited information about the target data, have been proposed at the intersection of privacy-preserving learning, generative modelling, and domain adaptation (more in Section 2). All these approaches, to our best knowledge, assume that the learner has access to the joint distribution of the target data, an assumption that the proposed approach relaxes significantly.

*equal contribution

Formally, ExPriL relies on two assumptions (Section 3.1). Firstly, an (unlabelled) source dataset is assumed to be available, as an i.i.d. sample drawn after a source distribution overlapping with the target data distribution. Secondly, the data owner, referred to as *Oracle* in the following, is willing to provide i) the marginal distribution of the target data; ii) the marginals of the errors committed by any submitted candidate hypothesis on the target data, under the ϵ-differential privacy requirements [DMNS06]. ExPriL starts by estimating the importance weights associated with each source sample, in order to match the marginal distribution of the target data. It thereafter iteratively builds a sequence of candidate hypotheses, and uses their marginal errors to estimate the target label associated to each sample.

This scheme can be viewed as an active learning scheme [BF13], with two differences. Firstly, the proposed learner asks for the fraction of errors attached to a bag of unknown target samples, whereas an active learning asks for the label attached to a known target sample. Secondly, an active learner selects the most informative known sample for its query, whereas the proposed approach only queries the oracle based on its current best hypothesis.

On the one hand, the joint target data distribution never leaves the private data curator, e.g. the hospitals or financials markets. On the other hand, the differential privacy of the mechanism [DMNS06] can be established through perturbing the marginals supplied by *Oracle* by addition of Laplacian noise. The empirical validation of the approach shows that the ExPriL differential privacy is obtained with a moderate loss of predictive accuracy for medium-dimensional problems (Section 4.3).

2 Related work

The presented approach is at the crossroads of privacy-preservation, generative modelling, and domain adaptation.

2.1 Privacy-preserving learning

A variety of real-world applications such as healthcare, customer analytics, financial reporting restrict the access to true data owing to privacy concerns. k-anonymisation for private data publishing works by blending a data point with $k-1$ (nearest) points to secure privacy, but suffers from attribute disclosure through homogeneity and inference attacks [MKGV07]; the problem becoming worse for high-dimensional data [Agg05]. Epsilon-differential privacy (ε-DP) is satisfied when a randomised statistical query f is able to ensure that for two datasets $\mathcal{D}$, $\mathcal{D}'$, that differ in a single entry alone, the responses are indifferentiable upto a factor of ϵ [DMNS06] i.e. $\mathcal{P}(f(\mathcal{D})) \leq e^{\epsilon}\mathcal{P}(f(\mathcal{D}'))$. This way, no adversary, with any amount of side information, can estimate the true value of any one data entry, thereby guaranteeing privacy of each and all entries.

The ε-DP definition has prompted a series of work on modelling synthetic datasets after a given target dataset through differentially private queries for data descriptions such as edge structure, conditional marginals [ZCP+17, PSH17]. Compared to [ZCP+17], for instance, our queries are simpler and only involve marginals along different feature axes.

Along a different line, PATE [PAUE+17] proposes the learning of a classification function through differentially private queries for the labels. Each data point is associated with a vector of label counts, obtained by aggregating the votes of an ensemble of teacher networks. The teachers are trained on disjoint subsets of the sensitive target dataset, for delimiting the label noise. Notice that this method precludes the creation of a synthetic dataset, modelled after the target, which could be useful for future, more diverse, analysis.

Then there are methods that learn in a federated manner from DP responses of individual data owners, under local differential privacy [KBR16, KOV14]. The lower bound on the averaging error for methods with perturbation to individual data points is larger than with perturbation to centralised responses [CSS12], the difference growing sub-linearly ($\sqrt{n}$) with the number of users (n). Our method sits between the two settings; although our data is gathered on a secure server, we have access only to the marginals of features, perturbed for privacy.

2.2 Generative modelling for sensitive data

Generative Adversarial Networks (GANs) [GPAM+14] have been demonstrated to learn probability distributions of data living in high-dimensional spaces. Sampling from the learned distribution provides an artificial dataset, that is viewed as a synthetic version of the true data. GANs are being heralded as a privacy-preserving solution to making sensitive data available in the public domain [JYS19, YDD+20]. However, deep neural networks are prone to data memorisation [ZBH+17]; the potential data leak among the target and the synthetic datasets is measured by the so-called *privacy loss* metric in *Health-GAN*. This measures the relative resemblance of the synthetic data to the real training and test target data respectively. In particular, the synthetic data distribution must be indistinguishable from both the training and test distributions for the privacy loss to be 0. While this metric (indirectly) captures the dissimilarity between synthetic and real samples, critiques [SOT20] question the alignment of this dissimilarity with formal/legal notions of privacy (i.e. how dissimilar is dissimilar enough?). PATE-GANs [JYS19] guarantee privacy by modelling their discriminators after the teacher architectures in PATE [PAUE+17], querying them for fake/real labels in a DP way. Contrarily, DPGAN [XLW+18], and DP-CGAN [TKP19] apply privacy perturbation to the gradients of discriminator training. Differential privacy is resistant to post-processing. This ensures that a generator trained from a DP-discriminator is itself differentially private.

More notably all the above methods depend on access to the target data, something that is unavailable to us.

2.3 Domain adaptation

In domain adaptation (DA) [BDBCP07, CFT], two different distributions $P_s(X,Y)$ and $P_t(X,Y)$ with X the sample description and Y its label, respectively referred to as source and target distributions, are considered. DA, usually assuming that $P_s(Y|X)$ is not too different from $P_t(Y|X)$, aims to exploit the wealth of source data to build a better model on the target domain, usually including little data and even less labels. DA approaches often rely on designing embeddings, mapping source and target instances on some latent space, such that i/ this embedding preserves the dis-

criminant information on the source data; ii/ it mixes the images of the source and target instances in such a way that the lack of target information is mitigated [GUA+16].

Extreme Domain Adaptation[1] exploits the source data together with minimal cues about the target data, expressed as the marginals of the label ($P(Y|X_i)$ with X_i a single descriptive feature). Another related approach is that of [HJKRR18], where a model is likewise adapted to achieve a calibrated prediction on all identifiable sub-groups within a given population.

In the privacy-preserving setting, we share closeness with [WGB19]. Their use of an intermediary dataset for transfer learning between private sources and target data requires a broader overlap assumption. The responses from sources, perturbed for privacy, are twofold, i/ source hypothesis, and ii/ an importance weight for each. Target distribution is seen as a convex combination of several source distributions, and the importance weights are learned by solving a system of linear equations. In [LHS19], the source data is assumed private, and queried for pairwise distances using Johnson-Lindenstrauss, and for labels using DP-histogram of ordered points. Our access to simply the marginals of private data sets us apart.

3 The ExPriL algorithm

ExPriL achieves extremely private supervised learning. The contribution of the approach is that it only requires the target data to be known from its marginals. In the following, we restrict ourselves to the binary label case and to d-dimensional instance spaces.

Let $\mathcal{D}_s = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ denote the source dataset, with $\mathbf{x}_i \in \mathbb{R}^d$. The domain of each feature is partitioned in q bins. Let $\mathcal{B} = \{B_1, \ldots B_K\}$ denote the set of bins. Note that the total number of bins $K = qd$ linearly grows with dimension d.

3.1 Overview

Formally, ExPriL relies on two assumptions:

1. The **target dataset** is drawn after some distribution P_t on the source domain, satisfying the overlap assumption w.r.t the source distribution ($P_t(A) > 0 \implies P_s(A) > 0$ with P_s the source distribution for all $A \subset \mathbb{R}^d$ subset of the instance space). No labelling of the source dataset is required.

2. The interaction with the *Oracle* provides the marginals of i) the distribution $P_t(X)$, that is, the mass contained in a feature bin; ii) the error of any candidate hypothesis submitted by ExPriL.

ExPriL iteratively addresses two subproblems: *source reweighting* (referred to as Pb. 1) and *label estimation* (Pb. 2).

Pb. 1 aims to associate an importance weight to each source data sample, such that the weighted source dataset matches to the best possible extent the target marginals provided by the oracle in response to the first query. Pb. 2 aims to estimate the (target) label associated to each source data sample, based on the error marginals provided by the oracle in response to each submitted candidate hypothesis. Along the differential privacy protocol all the marginals supplied by the *Oracle* are perturbed using Laplacian noise of adequate standard deviation (Section 3.5).

As will be shown below, both Pb. 1 and Pb. 2 can be formulated using a system of linear equations. For computational convenience, each system is solved by minimizing a convex optimization problem, using a stochastic gradient descent approach.

In the following, each sample is represented by concatenating the one-hot encodings associated to each feature, noting for each feature the bin it belongs to. Eventually, the dataset is encoded in binary (K, n) matrix R, where n is the number of samples, $K = qd$ is the total number of bins, and $R_{k,i} = 1$ iff sample i falls into bin k.[2] Defining the bins with regards to the target data directly would induce a privacy cost. Instead, the bins descriptions are optimized for the source dataset, and sent by the user alongside the candidate model, at each time step.

3.2 Pb. 1: Importance sampling

In this initial phase of the algorithm, one exploits the oracle output providing the marginals of the target dataset along every bin in $\mathcal{B}$. Formally, the *Oracle* yields $p_k = P_t(X \in B_k)$ for all bins in $\mathcal{B}$ (where p_k is perturbed by laplacian noise in the differentially private version). Let $\mathbf{p}$ denote the vector made of all p_k.

Let $\mathbf{w} \in \mathbb{R}^{+,n}$ denote the sought vector of importance weights associated to the source samples. Under the overlapping assumption and in the large sample limit, one has:

$$R\mathbf{w} = \mathbf{p} \ . \tag{1}$$

The search for the optimal $\mathbf{w}$ considers Eq. 1 augmented with a regularisation term aimed to avoid

[1] Uri Shalit, talk at ELLIS 2020

[2] More precisely, for $k = q\ell + j$, $R_{k,i} = 1$ iff the i-th sample falls in the j-th bin of the $(\ell + 1)$-th feature.

weight collapse and distribute the weights as equally as possible over all points in a bin. Letting $\mathbf{1}$ denote the 1-dimensional vector taking value 1 on every coordinate, $\mathbf{w}$ is sought as:

$$\mathbf{w} = \underset{\mathbf{w}\in\mathbf{R}^{+,n}}{\arg\min} \frac{1}{2}||R\mathbf{w} - \mathbf{p}||^2 + \alpha||\mathbf{w} - \frac{1}{n}\mathbf{1}||^2 \ . \quad (2)$$

The above equation defines a quadratic optimization problem, that can be handled using standard non-negative least squares optimization methods. As said, stochastic gradient descent is used for convenience.

In the following, by abuse of notation and for simplicity, R denotes the data matrix weighted with these importance weights ($R := Rdiag(\mathbf{w})$).

3.3 Pb. 2: Label Estimation

Any binary hypothesis h defined on $\mathbb{R}^d$ defines a label in $\{0,1\}$ for each data sample in the target data. We denote the vector of predictions as $\mathbf{h} = (h(x_1), \ldots, h(x_n))^t \in \{0,1\}^n$. By definition, letting $\mathbf{q}$ denote the vector yielding the fraction of errors of h compared to the ground truth label h^* on each bin in $\mathcal{B}$: for a given bin k, $q_k = \frac{\sum_{x in B_k} \delta_{h(x)\neq h^*(x)}}{\sum_{x in B_k} 1}$. With $\cdot$ the Hadamard product, and still under the overlapping and large sample limit assumptions, it comes:

$$R\ |\mathbf{h} - \mathbf{h}^*| = (R\ \mathbf{1}) \cdot \mathbf{q} \ . \quad (3)$$

This equality is handled as an approximation to account for the fact that the weighted source dataset only approximately matches the target dataset. A differentiable optimization objective is obtained by rewriting the above as:

$$R\ diag(sign(\mathbf{h} - \mathbf{h}^*))\ (\mathbf{h} - \mathbf{h}^*) \approx (R\ \mathbf{1}) \cdot \mathbf{q} \ . \quad (4)$$

The vector of prediction $\mathbf{h}$ is binary, so it follows that the ground truth $\mathbf{h}^*$ is necessarily solution to

$$R\ diag(sign(\mathbf{h} - 0.5))\ \mathbf{h}^* \approx -(R\ \mathbf{1}) \cdot \mathbf{q} + R\ \mathbf{h} \ . \quad (5)$$

Precisely, the ExPriL algorithm defines a sequence of candidate hypotheses h_t for $t = \{0 \ldots T\}$. Each h_t is submited to the oracle, yielding the vector $\mathbf{q}_t$ of errors on each bin in $\mathcal{B}$. The first hypothesis h_0 is set to the constant hypothesis predicting the label 1 for every sample in the target dataset.

Two ExPriL variants are considered:
Instantaneous-ExPriL (referred to as I-ExPriL) solves at iteration t the convex optimization problem derived from the current linear system (Eq. 5).

Cumulative-ExPriL (referred to as C-ExPriL) considers at iteration t the full stacked system with $K.t$ equations, stacking matrices $R\ diag(sign(\mathbf{h}_\ell - 0.5))$ for $\ell \leq t$ into a single matrix $\mathring{R} \in \mathcal{M}_{Kt,n}$ and the vectors $-(R\ \mathbf{1}) \cdot \mathbf{q}_\ell + R\ \mathbf{h}$ into a single vector $\mathring{\mathbf{q}}$. Likewise, an estimation $\hat{y}$ of the ground truth vector $\mathbf{h}^*$ is sought as solution of equation

$$\mathring{R}\mathbf{h}^* = \mathring{\mathbf{q}} \ . \quad (6)$$

This linear system with Kt equations and n unknowns is tackled as a convex optimization problem, relaxing the binary constraint and taking $\hat{y}$ in $[0,1]^n$, and minimizing the squared difference of the right and left hand sides by gradient descent.

3.4 Algorithm

The pseudo-code of ExPriL illustrates the estimation of labels $\hat{y}_i \in [0,1]$ of the weighted source samples, where hypothesis h is described by parameter vector θ.

Algorithm 1: Iterative marginals matching

```
Max. number of iterations T
Init: t := 0
    Query the Oracle:
      p = target binned marginals
    Learn w from Eq. 2
    h_0(X; θ := θ_0) s.t. h_0 = 1 everywhere
for t in [1, T] do
    Query the Oracle:
      q_{t-1} = error marginals incurred by h_{t-1}
    Update(R̊, q̊):
      [per I-ExPriL or C-ExPriL ]
    Solve h* = arg min||R̊ĥ − q̊||^2
    Learn h_t(θ := θ_t) s.t.
      θ_t = arg min_θ loss(h*; h_t)
end
```

Learning h_t. Let $\hat{y}_i$ be the (relaxed) label of sample i estimated at iteration $(t-1)$ (Eq. 6): it is mapped to $\{0,1\}$, and h_t is straightforwardly learned to minimize the cross-entropy loss from the labelled dataset $(\mathbf{x}_i, \hat{y}_i)$ for $i = 1 \ldots n$, with w_i the weight of the i-sample.

It is emphasized that, when the answers to the queries are perturbed by addition of Laplacian noise, the set of queries defines an ϵ-differentially private protocol.

3.5 ϵ-differential privacy

Following the differential privacy protocol defined by [DMNS06], an i.i.d. noise sampled from $Lap(\lambda)$ is added to each entry of the query feedback. Note that contrarily to common differential privacy routines, we add noise to the numerator only, rather than to the ratio itself. We apply *clip* operation to guarantee the differential privacy of the communication.

Lemma Let h be a binary model defined on a 1-dimensional instance space, $\mathcal{B} = \{B_1, \dots B_Q\}$ be a set of bins on this instance space, and $clip(x)$ be a function that returns 0 if x is negative, 1 if $x \geq 1$, x otherwise. A randomized algorithm $\mathcal{A}$ operates on a supervised dataset $\mathcal{D} = (x_i, y_i)_{i \in [\![1,n]\!]}$, and returns the clipped, noisy error-proportions $(C_1, C_2, \cdots, C_Q)$ of h in each bin $q \in \{1, 2, \cdots Q\}$ s.t.

$$C_q \sim clip\Big(\frac{\mathcal{L}aplace(\lambda) + \sum_{x_i \ in \ B_q} \delta_{h(x_i) \neq y_i}}{\sum_{x_i \ in \ B_q} 1}\Big). \quad (7)$$

Then the algorithm $\mathcal{A}$ is $\frac{1}{\lambda}$-differentially private.

Proof Let us consider two datasets $\mathcal{D}$ and $\mathcal{D}'$ such that $\mathcal{D} = \mathcal{D}' \bigcup \{x\}$, and let us assume with no loss of generality that x belongs to bin B_1. Let us assess the probability for $\mathcal{A}$ to return the same output for $\mathcal{D}$ and $\mathcal{D}'$. For $(c_1, \dots, c_Q)$ be in $[0,1]^Q$, let us define:

$$\exp(\eta) = \frac{\mathbb{P}(\mathcal{A}(\mathcal{D}) = (c_1, \dots, c_Q))}{\mathbb{P}(\mathcal{A}(\mathcal{D}') = (c_1, \dots, c_Q))} \ . \quad (8)$$

$\mathcal{A}$ is ϵ-DP if $\eta \leq \epsilon$ for any $(c_1, \dots, c_Q)$. By independence of the noise term in each bin,

$$\exp(\eta) = \frac{\mathbb{P}(\mathcal{A}(\mathcal{D})_1 = c_1)}{\mathbb{P}(\mathcal{A}(\mathcal{D}')_1 = c_1)} \quad (9)$$

Let us now suppose that $h(x) = y$ (same result holds if $h(x) \neq y$). We note $U = \sum_{x_i \ in \ B_1} \delta_{h(x_i) \neq y_i}$, $V = \sum_{x_i \ in \ B_1} \delta_{h(x_i) = y_i}$ and L the Laplacian noise term. Then,

$$\exp(\eta) = \frac{\mathbb{P}(clip(\frac{U+L}{U+V}) = c_1)}{\mathbb{P}(clip(\frac{U+L}{U+1+V}) = c_1)} \ . \quad (10)$$

At this point it appears that the clipping is necessary to ensure that the mechanism is differentially private :

$$\frac{\mathbb{P}(\frac{U+L}{U+V} = c_1)}{\mathbb{P}(\frac{U+L}{U+1+V} = c_1)} = e^{-\frac{1}{\lambda}(|c_1(U+V)-U|-|c_1(U+1+V)-U|)} \quad (11)$$

which, depending on c_1, might be greater than $exp(\frac{1}{\lambda})$.

Using the clipped formula, three cases arise. If $c_1 \in]-1, 1[$,

$$\exp(\eta) = \frac{\exp\Big(-\frac{1}{\lambda}(|c_1(U+V)-U|)\Big)}{\exp\Big(-\frac{1}{\lambda}(|c_1(U+1+V)-U|)\Big)} \leq \exp(\frac{1}{\lambda}) \ . \quad (12)$$

If $c_1 = 1$,

$$\exp(\eta) = \frac{\mathbb{P}(L > V)}{\mathbb{P}(L > V+1)} = \exp(\frac{1}{\lambda}) \ . \quad (13)$$

If $c_1 = 0$, the computations are identical.
Hence, $\mathcal{A}$ is $\frac{1}{\lambda}$-differentially private. □

In our setup, a single query corresponds to soliciting marginals along d dimensions simultaneously; a mechanism of querying marginal along a single dimension being similar to that of the lemma above. The composability property of differential privacy then ensures that each step/query is $\frac{d}{\lambda}$ differentially private.

Stacking $T+1$ such queries, one for learning the importance weights (Pb. 1) and one for each hypothesis $h_t \ \forall \ t \in \{0, \dots T-1\}$ (Pb. 2) yields a total privacy cost of at most $d(T+1)/\lambda$. Hence, setting

$$\lambda \geq d(T+1)/\epsilon, \quad (14)$$

ensures that ExPriL is ϵ-differentially private.

4 Experimental validation

This section begins by describing the benchmarks, then details the experimental setting and finally reports the results. The goal of the experiments is to assess (i) the performance of ExPriL compared to that of a reference baseline, trained directly on the target data, and (ii) to compare the instantaneous and cumulative versions of ExPriL, along with an evaluation of the sensitivity of the approach to the dimension d of the dataset.

4.1 Benchmarks

Besides a real-world problem (Cardiotocography dataset), five artificial problems noted A, B, C, D, E have been considered, with dimensions ranging in 2, 4, 10, 15, 25. The hyper-parameters of the generation process are described in Table 1.

Artificial datasets A and B Both the source and target instances distributions are sampled from a d-dimensional uniform law over the hypercube $[0,1]^n$. A randomly initialized K-Means algorithm clusters the target. Half of the clusters are randomly assigned the label 0, and the remaining the label 1. All samples inherit their label from the value of the cluster they belong to. Finally, the label of all target samples x_i that are the solution to the equation $\frac{d}{2} - \nu \leq \sum_{l=1}^{d} x_{i,l} \leq \frac{d}{2} + \nu$ are flipped. The value of ν is chosen such that this set corresponds to approximately one third of all target samples.

Artificial datasets C, D and E A source $\mathcal{Z}_s$ and target $\mathcal{Z}_t$ latent distributions are sampled from two different but overlapping mixtures of Gaussians in a latent space of dimension d_l. A randomly initialized neural network f then maps the latent space points to the instance space, yielding a source $\mathcal{D}_s = f(\mathcal{Z}_s)$ and a target $\mathcal{D}_t = f(\mathcal{Z}_t)$ distribution, of dimension d_i. Another randomly initialised neural network f' is defined, and the target labels are obtained by thresholding its output $f'(\mathcal{Z}_t)$.

Cardiotocography dataset The Cardio-Tocography (CTG) dataset [DG20] is a public medical dataset. After normalization and feature processing, two versions with 22 real-valued features are considered; CTG-A with same source and target instance distributions, and CTG-B where the two distributions are significantly different (but overlapping).

CTG-A: the original dataset is randomly split into a source and a target dataset ;
CTG-B: a specific feature, the heart beat rate, is considered. Depending on its value $f(x)$ normalized in [.1, .9]), sample x is selected as source sample with probability $f(x)$, otherwise, it is selected as target sample. Source and target distributions are thus different, though they satisfy the overlapping assumption. Source and target datasets have the same size.

4.2 Experimental setting

The hypothesis space is made of neural nets, with architectures described in Table 1. The (optimal) baseline is given by the average accuracy of a model h^* with the described neural architecture trained directly on the considered target dataset.

The target dataset is divided into two: the *target validation set* is used to compute the oracle feedback (marginals of the target distribution and error marginals of candidate hypotheses), the *target test set* is used to measure the reported performance of the approach.

The learning curve on each problem is reported, where the performance of I-ExPriL or C-ExPriL is reported relatively to the baseline performance. More precisely, the learning curve *Perf*(t) reports for iteration t the performance in terms of $(acc(h_t) - acc(\hat{h}^*))$.

Experiments are averaged over 100 independent runs by varying the split of the target dataset into *validation* (used to answer the queries) and *test* sets, and the pattern of perturbation of the target labels. Parameters and settings of each experiment are detailed in Table 1. The number of ExPriL queries (governing the Laplacian noise for differential privacy) is set to 3.

The bin descriptions are extracted from the source dataset, and consequently can be adapted at any stage in the pipeline. We define bins such that the distribution of sample mass is fairly uniform, while balancing the trade-off between the *precision* of the binning and the *signal-to-noise* ratio of the *Oracle* feedback: more precise bins may have lower signal-to-noise ratio. Considering q bins per feature sharing approximately evenly n' target samples, the expected number of samples per bin is equal to n'/q. The ratio of the number of samples in a bin to the noise standard deviation is thus close to $\frac{n'}{q\sqrt{2}\lambda}$, a value we keep close to 4 in the experiments. For Pb. 1 we base this calculation on the source distribution, and for Pb. 2 on the weighted source distribution.

4.3 Experimental results

In cases where the learner is aware that the source and target instance distributions are the same, the IS step (Pb. 1) can be skipped. In our experiments, this pertains to datasets A, B and CTG-A, for which we accordingly bypass the IS step. For dataset C, D, E, and CTG-B, the instance distributions are different, and hence IS step is retained. Table 1 details the experimental settings, with d_i the number of instance dimensions, d_l the number of latent dimensions, and n_t the number of target samples.

Note that the learning model is always a neural network. Table 2 reports the performance of the model learned from the validation target dataset $\hat{h}^*$, and that of the constant model predicting the majority class h_0^*. We report the performance of C-ExPriL (resp. I-ExPriL) in the differentially private version of the algorithm, with ϵ set to 1, under 1DP-C-ExPriL (resp. 1DP-I-ExPriL). The version without any privacy concern, where ϵ can be seen as extremely large, is denoted

Dataset	#runs	#queries (inc. IS step)	IS step	#target samples	Parameters	[Hidden Layers]
A	100	2	No	2500	15 clusters, $d_i = 2$	[16,256,256,16]
B	100	2	No	2500	64 clusters, $d_i = 4$	[16,256,256,16]
C	100	3	Yes	5000	$d_l = 3$, $d_i = 10$	[32,32]
D	100	3	Yes	5000	$d_l = 5$, $d_i = 15$	[32,32]
E	100	3	Yes	5000	$d_l = 10$, $d_i = 25$	[32,32]
CTG-A	100	2	No	700	n/a	[16,256,256,16]
CTG-B	100	3	Yes	700	n/a	[16,256,256,16]

Table 1: Settings of the different experiments

Dataset	$\hat{h}^*$	C-ExPriL	I-ExPriL	1DP-C-ExPriL	1DP-I-ExPriL	h_0^*	ratio
A	94.3 ± 2.3	76.8 ± 4.6	74.9 ± 5.0	74.0 ± 4.8	71.9 ± 4.6	56.2 ± 5.0	1.8e-1
B	79.9 ± 2.0	58.5 ± 2.5	57.4 ± 2.4	53.8 ± 2.7	53.5 ± 2.7	52.9 ± 2.3	2.7e-1
C	96.2 ± 1.5	88.8 ± 3.3	88.7 ± 3.8	85.8 ± 4.0	85.6 ± 4.0	55.6 ± 3.9	7.7e-2
D	90.9 ± 2.1	82.2 ± 3.5	80.7 ± 3.9	75.1 ± 4.4	74.1 ± 4.7	54.6 ± 2.8	9.5e-2
E	80.3 ± 2.5	72.3 ± 3.4	70.4 ± 3.7	63.0 ± 4.1	60.1 ± 3.9	54.0 ± 2.4	1.0e-1
CTG A	92.0 ± 1.0	87.8 ± 1.5	86.9 ± 1.6	73.0 ± 3.7	70.5 ± 3.2	77.7 ± 1.2	4.5e-2
CTG B	91.2 ± 1.1	86.1 ± 1.4	84.9 ± 1.8	70.1 ± 3.8	65.1 ± 4.0	75.5 ± 1.4	5.5e-2

Table 2: Comparative performances of both models (average and standard deviation over 100 runs)

by C-ExPriL (resp. I-ExPriL). The ratio indicator is evaluated as $\frac{acc(\hat{h}^*) - acc(ExPriL)}{acc(\hat{h}^*)}$, with $\hat{h}^*$ learned from the true target dataset.

General comments. As shown in Table 2, in all experiments except B and C and in both the no-DP and the 1-DP cases, C-ExPriL significantly outperforms I-ExPriL, with confidence over 95% after Wilcoxon-Mann-Whitney signed test. The learning curves (subsection 4.3) show that both C-ExPriL and I-ExPriL performance plateau after the first iteration.

As could have been expected the performances of the 1-DP algorithms are lower than that of their no-DP counterparts. The gap between those performances increases with number of dimensions of the instance space. This is explained by the fact that the level of differential noise λ is proportional to the number of dimensions d. Additionally, the performance of 1-DP version of both C-ExPriL and I-ExPriL falls below the baseline (h_0^*) for CTG dataset. A likely reason for this fact is that the level of differential noise is too high (compared to the signal) for the oracle feedback to be meaningful for CTG dataset, particularly in the CTG setting of higher dimensions combined with a low sample size ($n \sim 700$ compared to ~ 5000 in E).

Learning curves. For the learning curves of Figure 1, the averaged accuracy of a model h^* trained directly on the target dataset is taken as reference. A point in the graph at coordinate $[t,\ y]$ represents the performance of a model h after the t^{th} iteration where the performance is summarized in $y = acc(h) - acc(\hat{h}^*)$.

One sees from these learning curves that the performance plateaus very soon (except on dataset A); the magnitude of the Laplacian noise for guaranteeing privacy thus is over-dimensioned (expecting a total of 3 queries) and could have been reduced to 2.

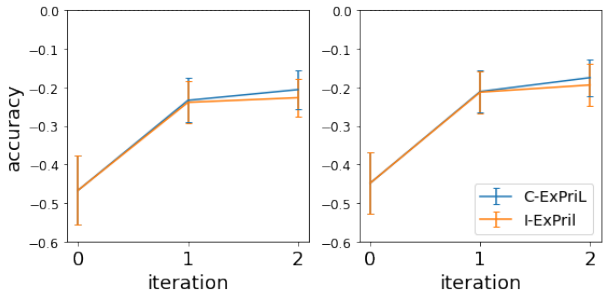
5 Discussion and Perspectives

The contribution of this paper has been to show how to learn from a target dataset that is never released by their owner, in a differentially private way. The main limitation of the approach is that the level of noise required to enforce DP increases with the dimensionality d of the instance space, and with the overall number of queries T.

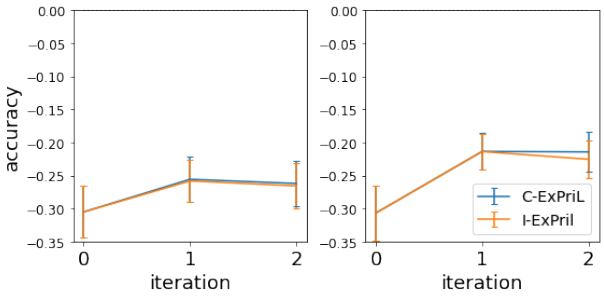
Further work is concerned with addressing this limitation, along the following ways. A first perspective is based on the remark that the queries for the error marginals could be distilled over the iterations. In basic mode, one could randomly subsample the features queries in each iteration, mechanically increasing the number of allowed iterations or decreasing the amplitude of the Laplacian noise. Along this same approach, one could select the most informative features (in terms of supervised feature selection, or considering the entropy of the errors in the bins). Finally, one could vary the bins to be queried in each iteration.

Another perspective consists in designing new informative features, and querying the oracle to provide the error marginals along these features.

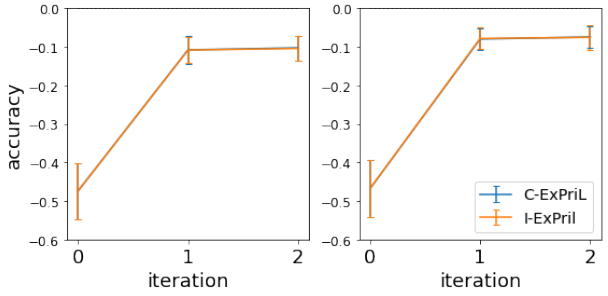
A third perspective is to extend the approach to the regression case, where the query would return the average squared error in each bin.



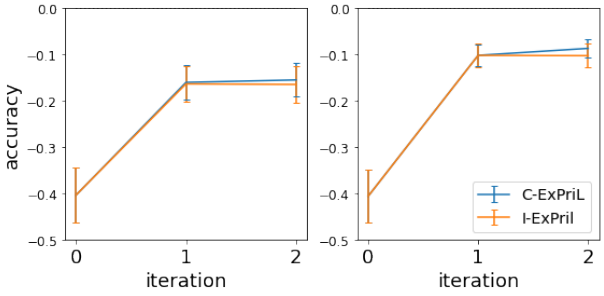
Dataset A, 1-DP (left), no-DP (right)



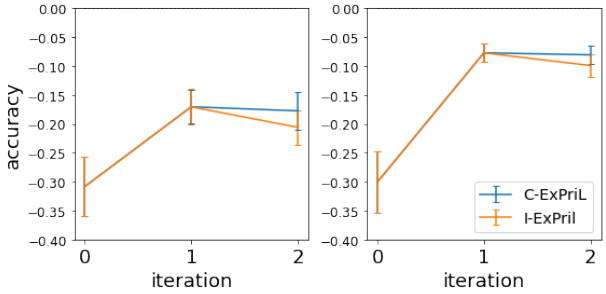
Dataset B, 1-DP (left), no-DP (right)



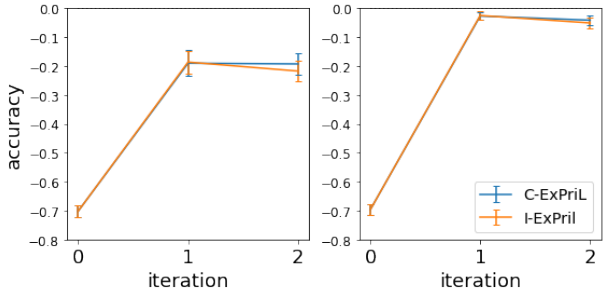
Dataset C, 1-DP (left), no-DP (right)



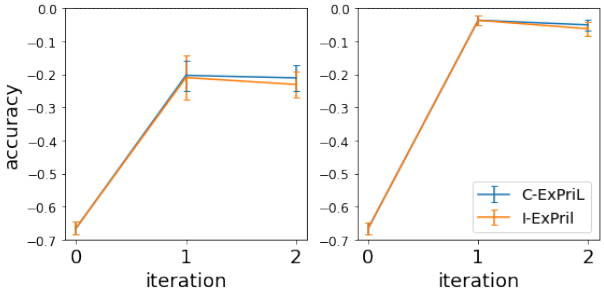
Dataset D, 1-DP (left), no-DP (right)



Dataset E, 1-DP (left), no-DP (right)



Dataset CTG-A, 1-DP (left), no-DP (right)



Dataset CTG-B, 1-DP (left), no-DP (right)

Figure 1: Learning curves

References

[Agg05] Charu Aggarwal. On k-anonymity and the curse of dimensionality. *VLDB 2005 - Proceedings of 31st International Conference on Very Large Data Bases*, 2005.

[BDBCP07] Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. In *Advances in Neural Information Processing Systems*, 2007.

[BF13] Maria-Florina Balcan and Vitaly Feldman. Statistical active learning algorithms. In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 1295–1303, 2013.

[CFT] Nicolas Courty, Rémi Flamary, and Devis Tuia. Domain adaptation with regularized optimal transport. In *ECML/PKDD 2014*.

[CSS12] T-H. Hubert Chan, Elaine Shi, and Dawn Song. Optimal lower bound for differentially private multi-party aggregation. In *Proceedings of the 20th Annual European Conference on Algorithms*, 2012.

[DG20] Dheeru Dua and Casey Graff. UCI machine learning repository, 2020.

[DMNS06] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the Third Conference on Theory of Cryptography*, 2006.

[GPAM+14] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, 2014.

[GUA+16] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.*, 2016.

[HJKRR18] Ursula Hebert-Johnson, Michael Kim, Omer Reingold, and Guy Rothblum. Multicalibration: Calibration for the (Computationally-identifiable) masses. In *Proceedings of the 35th International Conference on Machine Learning*, 2018.

[JYS19] J. Jordon, Jinsung Yoon, and M. V. D. Schaar. Pate-gan: Generating synthetic data with differential privacy guarantees. In *Proceedings of the International Conference on Learning Representations*, 2019.

[KBR16] Peter Kairouz, Keith Bonawitz, and Daniel Ramage. Discrete distribution estimation under local privacy. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, 2016.

[KOV14] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. Extremal mechanisms for local differential privacy. In *Advances in Neural Information Processing Systems*, volume 27, 2014.

[LHS19] Nam LeTien, Amaury Habrard, and Marc Sebban. Differentially private optimal transport: Application to domain adaptation. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, 2019.

[MKGV07] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. L-diversity: Privacy beyond k-anonymity. *ACM Trans. Knowl. Discov. Data*, 2007.

[PAUE+17] Nicolas Papernot, Martín Abadi, Úlfar Erlingsson, Ian Goodfellow, and Kunal Talwar. Semi-supervised knowledge transfer for deep learning from private training data. In *Proceedings of the International Conference on Learning Representations*, 2017.

[PSH17] Haoyue Ping, Julia Stoyanovich, and Bill Howe. Datasynthesizer: Privacy-preserving synthetic datasets. In *Proceedings of the 29th International Conference on Scientific and Statistical Database Management*, 2017.

[SOT20] Theresa Stadler, Bristena Oprisanu, and Carmela Troncoso. Synthetic data – a privacy mirage, 2020.

[TKP19] Reihaneh Torkzadehmahani, Peter Kairouz, and Benedict Paten. Dp-cgan: Differentially private synthetic data and

label generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2019.

[WGB19] Yang Wang, Quanquan Gu, and Donald Brown. Differentially private hypothesis transfer learning. In Michele Berlingerio, Francesco Bonchi, Thomas Gärtner, Neil Hurley, and Georgiana Ifrim, editors, *Machine Learning and Knowledge Discovery in Databases*, 2019.

[XLW+18] Liyang Xie, Kaixiang Lin, Shu Wang, Fei Wang, and Jiayu Zhou. Differentially private generative adversarial network, 2018.

[YDD+20] Andrew Yale, Saloni Dash, Ritik Dutta, Isabelle Guyon, Adrien Pavao, and Kristin P. Bennett. Generation and evaluation of privacy preserving synthetic health data. *Neurocomputing*, 416, 2020.

[ZBH+17] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *Proceedings of the International Conference on Learning Representations*, 2017.

[ZCP+17] Jun Zhang, Graham Cormode, Cecilia M. Procopiuc, Divesh Srivastava, and Xiaokui Xiao. Privbayes: Private data release via bayesian networks. *ACM Trans. Database Syst.*, 2017.