An Efficient Difference-of-Convex Solver for Privacy Funnel

Teng-Hui Huang and Hesham El Gamal
School of Electrical and Computer Engineering, University of Sydney
NSW, Australia
{tenghui.huang, hesham.elgamal}@sydney.edu.au

***Abstract*—We propose an efficient solver for the privacy funnel (PF) method, leveraging its difference-of-convex (DC) structure. The proposed DC separation results in a closed-form update equation, which allows straightforward application to both known and unknown distribution settings. For known distribution case, we prove the convergence (local stationary points) of the proposed non-greedy solver, and empirically show that it outperforms the state-of-the-art approaches in characterizing the privacy-utility trade-off. The insights of our DC approach apply to unknown distribution settings where labeled empirical samples are available instead. Leveraging the insights, our alternating minimization solver satisfies the fundamental Markov relation of PF in contrast to previous variational inference-based solvers. Empirically, we evaluate the proposed solver with MNIST and Fashion-MNIST datasets. Our results show that under a comparable reconstruction quality, an adversary suffers from higher prediction error from clustering our compressed codes than that with the compared methods. Most importantly, our solver is independent to private information in inference phase contrary to the baselines.**

I. Introduction

The privacy funnel (PF) method attracts increasing attention recently, driven by the need for enhanced data security as modern machine learning advances. However, due to the non-convexity of the involved optimization problem, the strict Markov constraint, and the requirement of full knowledge of a probability density, the PF problem is well-known to be difficult to solve. In the PF method, the goal is to solve the following optimization problem [1]:

$$\min_{P(Z|X)\in\Omega_{Z|X}} I(Z;Y), \quad \text{subject to } I(Z;X) \geq \eta_X, \tag{1}$$

where η_X denotes a control threshold. In the above problem, the joint distribution $P(X,Y)$ is assumed to be known and the random variables satisfy the Markov chain $Y \to X \to Z$. Here, Y is the private information; X the public information and Z is the released (compressed) codes; $I(Z;X)$ denotes the mutual information between two random variables Z and X. In (1), the variables to optimize with are the stochastic mappings $P(Z|X)$ with $\Omega_{Z|X}$ denotes the associated feasible solution set. The aim of the problem is to characterize the fundamental trade-off between the privacy leakage and the utility of the released information. One of the challenges in solving (1) is the non-convexity of the problem. It is well-known that the mutual information $I(Z;X)$ with known marginal $P(X)$ is a convex function with respect to $P(Z|X)$ [2]; Additionally, due to the Markov relation, $I(Z;Y)$ is a convex function of $P(Z|X)$ as well. Together, (1) minimizes a convex objective under a non-convex set. Previous solvers for PF address the non-convexity by restricting the feasible solution set further for efficient computation. Both [1] and [3] consider discrete settings and focus on deterministic $P(z|x)$, i.e., a realization of $x \in \mathcal{X}$ belongs to a single cluster $z \in \mathcal{Z}$ only. They propose greedy algorithms to solve the reduced problem and differ in choosing the subset of $\mathcal{Z}$ to combine with. However, the imposed restriction limits the characterization of the privacy-utility trade-off. In our earlier work [4], we propose a non-convex alternating direction methods of multipliers (ADMM) solver for PF which is non-greedy, but it suffers from slow convergence [5].

Another challenge of PF is knowing the joint distribution $P(X,Y)$. Overcoming this is important for practical application of the method as $P(X,Y)$ is prohibitively difficult to obtain in general. Nonetheless, it is relatively easy to obtain samples of the joint distribution, which become a training dataset as an indirect access to $P(X,Y)$. Recently, inspired by the empirical success of information theoretic-based machine learning, several works have attempted to extend the PF method through the theory of variational inference [6]. In [7], the conditional mutual information form of PF due to $I(Y;Z) = I(X;Z) - I(X;Z|Y)$ is adopted, whose variational surrogate bound is estimated with empirical samples. In [8] a similar formulation is adopted, but they restrict the parameter space and implement a generative deep architecture for interchangeable use of the empirical and the generated Y samples for estimation. However, owing to the dependency of the private information Y in estimating $I(X;Z|Y)$, these previous solvers violate the fundamental Markov relation in PF method. Consequently, samples of X, Y are required for their solvers which means supervision is required in both model training and inference phases.

Different from the previous works, we focus on more computationally efficient PF solvers for both known and unknown (empirical samples available) joint distribution settings based on the difference-of-convex (DC) structure of (1). In known joint distribution settings, inspired by the difference-of-convex algorithm (DCA), we propose a novel PF solver with closed-form update equation. In our evaluation, the proposed solver not only covers solutions obtained from the state-of-the-art

solvers but also attains non-trivial solutions that the compared greedy solvers are infeasible to reach. In unknown $P(X,Y)$ settings, our DC method complies with the fundamental Markov relation, therefore requires public information X only in inference phase in contrast to previous approaches. We implement our solver with deep neural networks and evaluate the model on MNIST and Fashion-MNIST datasets. In all evaluated datasets, our approach outperforms the state-of-the-art. Specifically, under a comparable reconstruction quality of the public information X, an adversary's clustering accuracy (leakage of the private information Y) with the released codes Z of ours is significantly lower than the compared approaches.

Notations: Upper-case letters denote random variables (RVs) and lower-case letters are their realizations. The calligraphic letter denotes the sample space, e.g., $x \in \mathcal{X}$ with cardinality $|\mathcal{X}|$. The vectors and matrices are denoted in bold-face symbols, e.g., $\boldsymbol{x}, \boldsymbol{X}$ respectively. We denote the conditional probability mass function as a long vector:

$$\boldsymbol{p}_{z|x} := \begin{bmatrix} P(x_1|z_1) & P(x_1|z_2) & \cdots & P(x_{|\mathcal{X}|}|z_{|\mathcal{Z}|}) \end{bmatrix}^T. \quad (2)$$

The Kullback-Leibler divergence of two measurable proper densities μ, ν is denoted as $D_{KL}[\mu \parallel \nu]$ [2].

II. A Difference-of-Convex Algorithm for Privacy Funnel

For simplicity, we focus on discrete random variables settings in this section. We start with a summary of DC programming and refer to the excellent review [9] for the development and recent advances. For a DC program, the canonical form is given by:

$$w^* = \arg\min_{w \in \mathcal{W}} f(w) - g(w), \quad (3)$$

where f, g are functions of w, and $\mathcal{W}$ denotes the feasible solution set of w. (3) relates to the PF method through rewriting (1) as an unconstrained PF Lagrangian:

$$\mathcal{L}_{PF} := I(Z;Y) - \beta I(Z;X), \quad (4)$$

where $\beta > 0$ is a multiplier. The above is a DC program when $P(X,Y)$ is known for a fixed β. This is because $I(Z;X)$ is convex w.r.t. $P(Z|X)$ for known $P(X)$, and $-H(Z|Y)$ is convex w.r.t. $P(Z|X)$ as the Markov relation $P(Z|Y) = \sum_{x \in \mathcal{X}} P(Z|x)P(x|Y)$ for known $P(Y)$ [2]. The DC structure of (4) can be leveraged for efficient optimization. By substituting the concave part with its first order approximation, we obtain a convex sub-objective. Then we iteratively minimize the convex surrogate sub-objective until the loss saturates. The monotonically decreasing loss values is assured due to the convexity of the sub-objective functions. However, the optimality of the converged solution is lost due to the non-convexity of the overall problem. There are well-developed efficient solvers for DC problems in literature, known as the Difference-of-Convex Algorithm (DCA). However, one of the key challenges in solving DC problems with DCA is that there exist infinitely many combinations of DC pairs [9]. Furthermore, a poorly chosen pair may result in slow convergence and numerical instability. We address the issues by proposing the following DC separation:

$$f(\boldsymbol{p}_{z|x}) := -H(Z|Y), \quad (5a)$$

$$g(\boldsymbol{p}_{z|x}) := -H(Z) + \beta I(Z|X), \quad (5b)$$

Then we apply (5) to DCA:

$$\boldsymbol{p}_{z|x}^{k+1} := \arg\min_{\boldsymbol{p}_{z|x} \in \Omega_{z|x}} f(\boldsymbol{p}_{z|x}) - \langle \nabla g(\boldsymbol{p}_{z|x}^k), \boldsymbol{p}_{z|x} - \boldsymbol{p}_{z|x}^k \rangle, \quad (6)$$

where the superscript k denotes the iteration counter. For known $P(X,Y)$ in discrete settings, (6) can be expressed as a linear equation (see Appendix A):

$$\boldsymbol{B}_{y|x} \log \boldsymbol{p}_{z|y} = \log \boldsymbol{p}_z^k + \beta \log \frac{\boldsymbol{p}_{z|x}^k}{\boldsymbol{p}_z^k}, \quad (7)$$

where $\boldsymbol{B}_{y|x} := \boldsymbol{I}_{|\mathcal{Z}|} \otimes \boldsymbol{Q}_{y|x}$, with $\boldsymbol{I}_d$ denotes the d-dimensional identity matrix, $\otimes$ the Kronecker product, and $\boldsymbol{Q}_{y|x}$ the matrix-form of $P(Y|X)$ whose (i,j)-entry is $P(y_i|x_j)$. We follow the assumption in private source coding literature that $\boldsymbol{B}_{y|x}$ is full-row rank [10]–[14]. Then we denote the Moore-Penrose pseudo-inverse of $\boldsymbol{B}_{y|x}$ as $\boldsymbol{B}_{y|x}^\dagger := (\boldsymbol{B}_{y|x}^T \boldsymbol{B}_{y|x})^{-1} \boldsymbol{B}_{y|x}^T$. Note that due to the Markov relation, we have $\boldsymbol{A}_{x|y} \boldsymbol{p}_{z|x} = \boldsymbol{p}_{z|y}$ with $\boldsymbol{A}_{x|y} := \boldsymbol{I}_{\mathcal{Z}} \otimes \boldsymbol{Q}_{x|y}$ defined similarly as $\boldsymbol{B}_{y|x}$. Summarizing the above, (7) reduces to:

$$\tilde{\boldsymbol{p}}_{z|y} \propto \exp\{\boldsymbol{B}_{y|x}^\dagger \boldsymbol{c}_{z|x}^k\}, \quad (8)$$

where $\boldsymbol{c}_{z|x}^k := \log \boldsymbol{p}_z^k + \beta \log \frac{\boldsymbol{p}_{z|x}^k}{\boldsymbol{p}_z^k}$, $\tilde{\boldsymbol{p}}_{z|y}$ denotes the unnormalized probability vector and $\propto$ represents proportional to. After normalization, followed by the Markov relation, the insight from applying DCA to PF gives an update equation:

$$\boldsymbol{A}_{x|y} \boldsymbol{p}_{z|x} = \boldsymbol{\phi}(\boldsymbol{B}_{y|x}^\dagger \boldsymbol{c}_{z|x}^k) := \boldsymbol{q}_{z|y}^k, \quad (9)$$

where the $z \in \mathcal{Z}$ element is given by $\phi_z = e^z / \sum_{z' \in \mathcal{Z}} e^{z'}$. In practice, $\boldsymbol{\phi}$ is implemented as a softmax function. The above implies that the update of $\boldsymbol{p}_{z|x}$ from step k to step $k+1$ requires solving the linear equation (9). However, since $\boldsymbol{A}_{x|y}^\dagger$ might not exist ($\boldsymbol{B}_{y|x}$ is full-row rank), (9) is an under-determined linear program over non-negative simplice. Heuristically, one would prefer a solution with the lowest complexity, which is often achieved by imposing q-norm (e.g., $q \in \{0,1,2\}$) constraints as regularization [15]. Based on this intuition, we therefore solve the following sub-problem:

$$\min_{\boldsymbol{p}_{z|x} \in \Omega_{z|x}} \|\boldsymbol{p}_{z|x}\|_q, \quad \text{subject to } \boldsymbol{A}_{x|y} \boldsymbol{p}_{z|x} = \boldsymbol{q}_{z|y}^k, \quad (10)$$

where $\|\boldsymbol{v}\|_q := (\sum_{v \in \mathcal{V}} |v|^q)^{1/q}$. Similar to relaxing the "hard" constraints in classical regression problems for applying more efficient algorithms, we solve the following relaxed problem of (10) where a soft constraint is considered instead:

$$\boldsymbol{p}_{z|x}^{k+1} := \arg\min_{\boldsymbol{p}_{z|x} \in \Omega_{z|x}} \frac{1}{2} \|\boldsymbol{A}_{x|y} \boldsymbol{p}_{z|x} - \boldsymbol{q}_{z|y}^k\|_2^2 + \alpha \|\boldsymbol{p}_{z|x}\|_q, \quad (11)$$

where $\alpha > 0$ is a relaxation coefficient. The relaxed problem tolerates an approximation error of the linear equation instead

of a strict equality constraint. Remarkably, (10) relates to the classical regression problems such as Ridge regression [16], LASSO [17], and sparse recovery [18]. But in (10) and (11), the feasible solution set $\Omega_{z|x}$ is the probability simplice. For $q = 2$, since (11) corresponds to a convex optimization problem, each step can be solved efficiently with off-the-shelf ridge regression solvers [19]. As for $q = 1$ (convex surrogate for $q = 0$), however, the LASSO solver [17] does not work well. This is because $\|\boldsymbol{p}_{z|x}\|_1 = \sum_{x\in\mathcal{X}} |\boldsymbol{p}_{Z|x}| = |\mathcal{X}|$, since $\sum_{z\in\mathcal{Z}} P(z|x) = 1, \forall x \in \mathcal{X}$ always. To address this, we provide the following alternative for sparse recovery ($q = 0$) with details referred to Appendix B:

$$l_{z|x}^{k+1} := \underset{l_{z|x}\in\mathcal{H}_{z|x}}{\arg\min} \frac{1}{2}\|lse_x(l_{x|y} + l_{z|x}) - l_\phi^k\|^2 + \alpha\|l_{z|x}\|_1, \quad (12)$$

where $l_{x|y} := \log \boldsymbol{p}_{x|y}$, $lse(x) := \log \sum_{x\in\mathcal{X}} \exp\{x\}$ is the log-sum-exponential function; $\mathcal{H}_{z|x}$ is the feasible set for $l_{z|x} \in [-M, -m]$ for some $M > m > 0$ such that $\sum_{z\in\mathcal{Z}} \exp\{l_{z|x}\} = 1, \forall x \in \mathcal{X}$; the subscript of lse denotes the variable for summation, and $l_\phi^k := \log \boldsymbol{\phi}(\boldsymbol{B}_{y|x}^\dagger \boldsymbol{c}_{z|x}^k)$. Then after solving (12), we can project the obtained $l_{z|x}$ to a feasible solution $\boldsymbol{p}_{z|x}$ through a softmax function: $\boldsymbol{p}_{z|x} = \text{Softmax}(l_{z|x})$.

For both implementations, the proposed solver guarantees convergence to a local stationary point.

Theorem 1: For both implementations (11) ($q = 2$) and (12), the sequence $\{\boldsymbol{A}_x \boldsymbol{p}_{z|x}^k\}_{k\in\mathbb{N}}$, obtained from (6), converges to a stationary point $\boldsymbol{A}_x \boldsymbol{p}_{z|x}^*$ such that $\nabla f(\boldsymbol{p}_{z|x}^*) = \nabla g(\boldsymbol{p}_{z|x}^*)$ with f, g defined in (5), and $\boldsymbol{A}_x := \boldsymbol{I}_\mathcal{Z} \otimes \boldsymbol{Q}_x$.

Proof: See Appendix D. ■

III. Extension to Unknown Distributions

The main difficulty in applying PF is the full knowledge of the joint distribution $P(X, Y)$ [1]. In (11), this corresponds to defining the operators $\boldsymbol{A}_{x|y}$ and $\boldsymbol{B}_{y|x}^\dagger$. Without knowing $P(X, Y)$, these operators are intractible. To address this, we leverage the DC structure of the PF problem again, but now take expectation with respect to $P(Z, X) = P(Z|X)P(X)$ on the update equation (see Appendix C):

$$I(Z;Y) + D_{KL}[P_z \parallel P_z^k] - \beta\mathbb{E}_{z,x}\left[\log P^k(X|Z)\right] = 0, \quad (13)$$

where the superscript k denotes the iteration counter. The above update equation implies that the step $k + 1$ solution $P^{k+1}(Z|X)$ is obtained from solving (13), given the step k solution $P^k(Z|X)$.

The expectation form (13) is useful because the computationally prohibitive knowledge of $P(X, Y)$ can be approximated efficiently through the theory of the variational inference, followed by Monte-Carlo sampling [6], [20]. The derived surrogate bound involves an auxiliary variable, which corresponds to the released information Z, and is associated with a variational distribution to be designed. For the mutual information $I(Z; Y) = H(Y) - H(Y|Z)$:

$$H(Y|Z) \le \mathbb{E}_{y,z;\phi}\left[\log \frac{1}{Q_\phi(Y|Z)}\right], \quad (14)$$

where the above surrogate bound is tight when $Q_\phi(Y|Z) = P(Y|Z)$. As for the other terms in (13), we parameterize $P_\theta(Z) = \mathcal{N}(\boldsymbol{0}, \text{diag}(\boldsymbol{\sigma}_\theta^2))$ as standard Gaussian and $P_\phi(X|Z) = \mathcal{N}(\boldsymbol{\mu}_\phi(\boldsymbol{z}), \boldsymbol{I})$ as conditional Gaussian distributions. This transforms the intractible problem into standard parameter estimation. Also, we parameterize the encoder $P_\theta(Z|X)$, corresponding to the expectation operator in (13), through the evident lower bound technique as adopted in the variational autoencoders (VAE) [20]. Here, we have $P_\theta(Z|X) \sim \mathcal{N}(\boldsymbol{\mu}_\theta(\boldsymbol{x}), \boldsymbol{\Sigma}_\theta(\boldsymbol{x}))$. Combining the above, we propose the following alternating solver:

$$\begin{aligned}\phi^* := \underset{\phi\in\Phi}{\arg\min} & -\beta\mathbb{E}_{x,z;\theta^k}\left[\log P_\phi(X|Z)\right] \\ & -\mathbb{E}_{y,z;\theta^k}\left[\log Q_\phi(Y|Z)\right], \quad (15a)\\ \theta^{k+1} := \underset{\theta\in\Theta}{\arg\min} & \frac{1}{2}\|I_{\phi^*;\theta}(Z;Y) + D_{KL}[P_{z;\theta} \parallel P_{z;\theta^k}] \\ & -\beta\mathbb{E}_{z,x;\theta}\left[\log P_{\phi^*}(X|Z)\right]\|_2^2 \\ & +\alpha\mathbb{E}_x\left[D_{KL}[P_{z|x;\theta} \parallel r_Z]\right], \quad (15b)\end{aligned}$$

where $r_Z = \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ is a reference probability density function, regularizing the encoder; and $\alpha > 0$ is a control threshold. Essentially, we alternate between fitting the empirical samples of X, Y (15a) and solving the DCA update sub-problem with an extra regularization term (15b). The two steps are repeated until a pre-determined number of iterations is reached.

The optimization of (15) is efficient. Each of the two KL divergence terms are computed between two Gaussian densities which has closed-form expression [21]; the parameterized $P_\phi(X|Z)$ and $Q_\phi(Y|Z)$ (hence $I(Z;Y) \approx H(Y) - H_\phi(Y|Z)$) can be estimated with mean-squared error (MSE) and a categorical cross entropy loss respectively. Finally, two remarks are in order:

- While variational inference is adopted of the proposed solver, but we apply it on the expectation form of the proposed DC update equation (13), not on the PF Lagrangian nor on $I(X; Z|Y)$ [7], [8].
- The parameterization of (15) can be extended to other density functions such as Laplace, Bernoulli and exponential distributions [22], [23].

IV. Evaluation

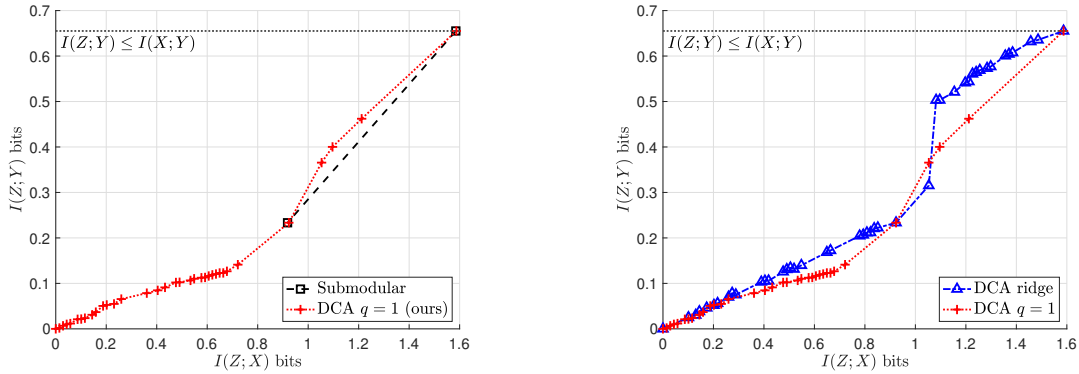
A. Known Distribution

The joint distribution we consider in this part is given by:

$$\boldsymbol{P}_{Y|X} = \begin{bmatrix} 0.90 & 0.08 & 0.40 \\ 0.025 & 0.82 & 0.05 \\ 0.075 & 0.10 & 0.55 \end{bmatrix}, \quad \boldsymbol{p}_X = \begin{bmatrix} \frac{1}{3} \\ \frac{1}{3} \\ \frac{1}{3} \end{bmatrix}. \quad (16)$$

Note that in (16), y_0, y_1 are easy to infer if observing x_0, x_1 whereas there is ambiguity in inferring y_0, y_2 if x_2 is observed instead.

The goal of our evaluation is to characterize the privacy-utility trade-off by plotting the lowest achieved $I(Z; Y)$ for an obtained $I(Z; X)$. The characterized trade-off is displayed on the information plane [1], [4], which depends on both $P(X, Y)$ and the algorithm used. We evaluate our DCA solver (5)



(a) DCA (12) with $q = 1$ versus *Submodular* [3]

(b) DCA (12) with $q = 1$ versus (11) with $q = 2$ (ridge)

Fig. 1. Comparing the characterization of the privacy-utility trade-off between the proposed DCA solvers and the state-of-the-art method [3]. The joint distribution $P(X, Y)$ is known and is given by (16).

and compare it to the state-of-the-art solvers [1], [3]. These baselines only consider deterministic $P(z|x) = \mathbf{1}\{x \in z\}$, that is, z categorizes $x, \forall x \in X$. These greedy algorithms merge two or more $z \in \mathcal{Z}$ iteratively for lower PF Lagrangian loss. Contrary to them, our solver is non-greedy with feasible solutions cover the (smoothed) probability simplice. For these baselines, it suffices to compare to [3] since [1] is a special case.

In the proposed solver (5), the range of cardinality $|\mathcal{Z}|$ follows the Carathéodory Theorem, i.e., $|\mathcal{Z}| \leq \max\{|\mathcal{X}|, |\mathcal{Y}|\} + 1$ [24]. For each cardinality, we initialize $P(Z|X)$ by randomly sample a uniformly distributed source in $[0, 1]$, followed by normalization to obtain a starting point. Then, we fix a trade-off parameter $\beta \in [0.1, 10.0]$ and a regularization coefficient $\alpha \in [0.1, 10.0]$, where each set consists of 16 geometrically spaced points. For each pair of hyperparameters, we run our solver until either the consecutive loss values satisfy $|\mathcal{L}^k - \mathcal{L}^{k+1}| \leq 10^{-6}$ or a maximum number of iterations $K = 10^4$ is reached. This procedure is repeated for 10 times. We compute the obtained metrics $I(Z; X), I(Z; Y)$ offline.

In Fig. 1a, our solver (5) characterizes the trade-off better than the baseline solvers since we not only cover their solutions but achieve more non-trivial points that are infeasible to them. In Fig. 1b we compare the two proposed DCA solvers with different implementations, i.e., (11) with $q = 2$ (ridge) versus (12) with $q = 1$. While the ridge approach performs worse than the other solver in characterizing the trade-off, it obtains a better trade-off point at $I(Z; X) \approx 1$ bit and $I(Z; Y) \approx 0.3$ bits. Moreover, the ridge solver benefits from the optimized off-the-shelf solver [19], hence converges faster in our empirical evaluation. It is worth noting that both approaches outperform [3] in characterizing the trade-off under different regularization criterion, demonstrating the strength of our DCA approach.

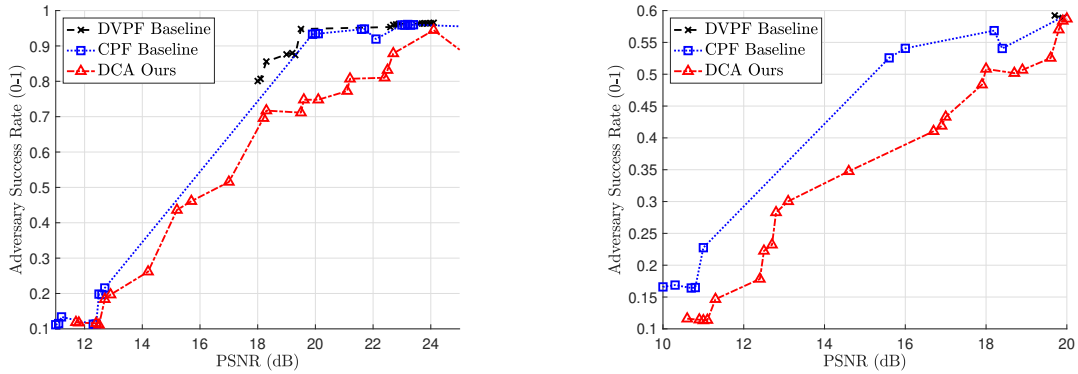
B. Unknown Distribution

For unknown $P(X, Y)$, we consider cases where empirical samples of the joint distribution are available (datasets) $\mathcal{D}_N := \{(x^{(n)}, y^{(n)}) | (x^{(n)}, y^{(n)}) \sim P(X, Y)\}_{n=1}^N$. Our DCA solver is compared to the state-of-the-art Conditional Privacy Funnel (CPF) [7] and Deep Variational Privacy Funnel (DVPF) [8] baselines.

a) Datasets: we evaluate our solver (15) on MNIST [25] and Fashion MNIST [26] datasets. Both datasets consist of gray-scale images of 28×28 pixels. Each pixel is normalized to the range $[0, 1]$. There are labeled 60000 training and 10000 testing samples uniformly distributed between 10 classes for each dataset. Specifically, the MNIST dataset consists of images of hand-written digits. Here, we let the digits Y be the private information and the images are treated as the public information X. As for Fashion MNIST, it has 10 categories of clothing. We refer to [26] for details. We set the categories as Y and the gray-scale images as X.

We consider the following scenario: a sender learns (supervised) an encoder-decoder pair with training dataset based on the PF objective. The decoder is shared to a receiver through a secure channel before inference phase. Then the sender encodes X of the testing set as the released codes Z. A receiver observes the coded Z and reconstruct $\hat{X}$ with the decoder $P_\phi(X|Z)$. Meanwhile, an adversary also observes the codes but does not know $P_\phi(X|Z)$. The adversary therefore learns $\hat{Y}$ from clustering Z. Here, we assume the cardinality of $|Y|$ is known for simplicity (relaxed with cluster identification algorithms [27]). The legitimate users' goal is to reconstruct $\hat{X}$ from Z whereas the adversary clusters Z as $\hat{Y}$ that should reveal Y.

In our evaluation, we measure the reconstruction quality (utility) by the Peak Signal-to-Noise power Ratio (PSNR), defined as:$PSNR := 10 \log X_{\max}/MSE$, where $X_{\max}$ is the maximum possible pixel value (1 in our case) and $MSE := \frac{1}{N}\sum_{i=1}^N \|\boldsymbol{x}_i - \hat{\boldsymbol{x}}\|_2^2$. As for the privacy leakage, the adversary uses UMAP [28] to project Z to a $\mathbb{R}^2$ space, followed by



(a) MNIST (b) Fashion-MNIST

Fig. 2. Privacy-utility trade-off in unknown joint distribution settings. The proposed DCA solver is compared to the CPF [7] and DVPF [8] baselines.

KMeans clustering [19]. Note that this two-step approach achieves the state-of-the-art clustering performance for both MNIST and Fashion-MNIST datasets [28]. Then we compute the adversary's success rate from offline label matching.

b) Network Architecture: The proposed and compared methods are implemented as deep neural networks. Since our focus is on comparing the PF objective, the encoders and decoders are carefully aligned for all compared methods. We implement the encoder as a neural network with fully connected (FC) layers as the standard VAE [20] encoder. We have a separate set of parameters for learning *a priori* $P_\theta(Z) = \mathcal{N}(\mathbf{0}, \boldsymbol{\nu}_\theta)$. As for the decoder, we employ another neural network with FC layers as the reconstruction module. For the differences, our method has a separate linear classifier whereas the CPF baseline expands the decoder's input by concatenating code samples $\boldsymbol{z}$ with the one-hot representation of target variables, i.e., $\boldsymbol{e}_y$ ($\boldsymbol{e}_i$ is the ith elementary vector). For the DVPF baseline, extra adversarial generative networks are implemented according to [8].

We train each model for 100 epochs with a learning rate at 3×10^{-4} using a standard ADAM optimizer [29]. The mini-batch size is 256. For the dimension of Z, we vary $d_z \in \{32, 64, 128, 256\}$ for all methods. The range of hyperparameters for our method is set to $[5 \times 10^{-4}, 5 \times 10^{-2}]$ for both α and β in (15), whereas it is $[10^{-3}, 10]$ for the CPF and DVPF baselines. Each hyperparameter pair is trained for 3 times with a different random seed and we report the average in the results. The experiment details are included in Appendix E. For reproduction, our source codes in available online[1].

c) Results: In Fig. 2, we report the highest adversary success rate of an achieved PSNR for each method. Note that the results include all combinations of the hyperparameters and the set of d_z. Fig. 2a corresponds to the MNIST dataset whereas the results for Fashion-MNIST dataset is shown in Fig. 2b. In both figures, our method clearly outperforms the CPF and DVPF baselines since the adversary has lower success rate in clustering Z from our approach at a comparable reconstruction quality of X. This demonstrates that our approach characterizes the privacy-utility trade-off better than both baselines. In addition to better robustness, our approach is more efficient in training phase than the DVPF baseline since we require no parameters for generative and discriminative modules whose optimization follows by a complex six-step algorithm. As for in inference phase, our approach is completely independent to the private information, in sharp contrast to the CPF baseline. For extended evaluation and visualization of the compressed features, we refer to Appendix F and Appendix G for details.

[1]available at: https://github.com/hui811116/dcaPF-torch

V. Conclusions

We leverage the DC structure of the PF method and develop efficient solvers based on DCA. In known distribution settings, our solver resembles both ridge regression and sparse recovery. Empirically, our solver outperforms the state-of-the-art greedy solvers in characterizing the privacy-utility trade-off. As for the unknown distribution settings, the insights from the proposed DC approach complies the fundamental Markov relation, so our approach applies to scenarios without labeled private information after training in contrast to compared methods. The evaluation of the proposed solver in MNIST and the Fashion-MNIST datasets demonstrates significantly improved characterization of the privacy-utility trade-off over the compared methods. For future work, we empirically find that the compression ratio ($|\mathcal{Z}|/|\mathcal{X}|$), significantly affects the characterization of the privacy-utility trade-off. A fundamental study in this direction will be the focus of a follow-up work.

References

[1] A. Makhdoumi, S. Salamatian, N. Fawaz, and M. Médard, "From the information bottleneck to the privacy funnel," in *2014 IEEE Information Theory Workshop (ITW 2014)*, pp. 501–505, 2014.

[2] T. M. Cover, *Elements of information theory*. John Wiley & Sons, 1999.

[3] N. Ding and P. Sadeghi, "A submodularity-based clustering algorithm for the information bottleneck and privacy funnel," in *2019 IEEE Information Theory Workshop (ITW)*, pp. 1–5, 2019.

[4] T.-H. Huang, A. E. Gamal, and H. E. Gamal, "A linearly convergent Douglas-Rachford splitting solver for Markovian information-theoretic optimization problems," *IEEE Transactions on Information Theory*, vol. 69, no. 5, pp. 3372–3399, 2023.

[5] H. Attouch, J. Bolte, P. Redont, and A. Soubeyran, "Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the Kurdyka-łojasiewicz inequality," *Mathematics of operations research*, vol. 35, no. 2, pp. 438–457, 2010.

[6] A. A. Alemi, I. Fischer, J. V. Dillon, and K. Murphy, "Deep variational information bottleneck," *arXiv preprint arXiv:1612.00410*, 2016.

[7] B. Rodríguez-Gálvez, R. Thobaben, and M. Skoglund, "A variational approach to privacy and fairness," in *2021 IEEE Information Theory Workshop (ITW)*, pp. 1–6, IEEE, 2021.

[8] B. Razeghi, P. Rahimi, and S. Marcel, "Deep variational privacy funnel: General modeling with applications in face recognition," *arXiv preprint arXiv:2401.14792*, 2024.

[9] H. A. Le Thi and T. Pham Dinh, "Dc programming and dca: thirty years of developments," *Mathematical Programming*, vol. 169, no. 1, pp. 5–68, 2018.

[10] F. P. Calmon, A. Makhdoumi, and M. Médard, "Fundamental limits of perfect privacy," in *2015 IEEE International Symposium on Information Theory (ISIT)*, pp. 1796–1800, 2015.

[11] C. Schieler and P. Cuff, "Rate-distortion theory for secrecy systems," *IEEE Transactions on Information Theory*, vol. 60, no. 12, pp. 7584–7605, 2014.

[12] K. Kittichokechai and G. Caire, "Privacy-constrained remote source coding," in *2016 IEEE International Symposium on Information Theory (ISIT)*, pp. 1078–1082, 2016.

[13] Y. Yakimenka, H.-Y. Lin, E. Rosnes, and J. Kliewer, "Optimal rate-distortion-leakage tradeoff for single-server information retrieval," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 3, pp. 832–846, 2022.

[14] Y. Y. Shkel, R. S. Blum, and H. V. Poor, "Secrecy by design with applications to privacy and compression," *IEEE Transactions on Information Theory*, vol. 67, no. 2, pp. 824–843, 2021.

[15] D. L. Donoho, "Compressed sensing," *IEEE Transactions on information theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[16] N. R. Draper and F. Pukelsheim, "Generalized ridge analysis under linear restrictions, with particular applications to mixture experiments problems," *Technometrics*, vol. 44, no. 3, pp. 250–259, 2002.

[17] B. R. Gaines, J. Kim, and H. Zhou, "Algorithms for fitting the constrained lasso," *Journal of Computational and Graphical Statistics*, vol. 27, no. 4, pp. 861–871, 2018.

[18] M. Pilanci, L. Ghaoui, and V. Chandrasekaran, "Recovery of sparse probability measures via convex programming," *Advances in Neural Information Processing Systems*, vol. 25, 2012.

[19] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[20] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[21] J. Duchi, "Derivations for linear algebra and optimization," *Berkeley, California*, vol. 3, no. 1, pp. 2325–5870, 2007.

[22] Y. Park, C. Kim, and G. Kim, "Variational Laplace autoencoders," in *International conference on machine learning*, pp. 5032–5041, PMLR, 2019.

[23] T.-H. Huang, T. Dahanayaka, K. Thilakarathna, P. H. Leong, and H. El Gamal, "The Wyner variational autoencoder for unsupervised multi-layer wireless fingerprinting," in *GLOBECOM 2023 - 2023 IEEE Global Communications Conference*, pp. 820–825, 2023.

[24] S. Asoodeh and F. P. Calmon, "Bottleneck problems: An information and estimation-theoretic view," *Entropy*, vol. 22, no. 11, p. 1325, 2020.

[25] Y. LeCun, C. Cortes, and C. Burges, "Mnist handwritten digit database," *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, vol. 2, 2010.

[26] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017.

[27] S. Ray and R. H. Turi, "Determination of number of clusters in k-means clustering and application in colour image segmentation," in *Proceedings of the 4th international conference on advances in pattern recognition and digital techniques*, vol. 137, p. 143, Calcutta, India:, 1999.

[28] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," *arXiv preprint arXiv:1802.03426*, 2018.

[29] D. P. Kingma and J. Ba, "ADAM: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[30] T.-H. Huang and H. El Gamal, "Efficient alternating minimization solvers for Wyner multi-view unsupervised learning," in *2023 IEEE International Symposium on Information Theory (ISIT)*, pp. 707–712, 2023.

[31] T.-H. Huang and H. E. Gamal, "Efficient solvers for Wyner common information with application to multi-modal clustering," *arXiv preprint arXiv:2402.14266*, 2024.

[32] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.