This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/MIS.2020.2993966, IEEE Intelligent Systems

Federated Generative Privacy

Aleksei Triastcyn
Artificial Intelligence Lab
Ecole Polytechnique Fédérale de Lausanne
Lausanne, Switzerland

Boi Faltings
Artificial Intelligence Lab
Ecole Polytechnique Fédérale de Lausanne
Lausanne, Switzerland

***Abstract*—We propose `FedGP`, a framework for privacy-preserving data release in the federated learning setting. We use generative adversarial networks, generator components of which are trained by `FedAvg` algorithm, to draw private artificial data samples and empirically assess the risk of information disclosure. Our experiments show that `FedGP` is able to generate labelled data of high quality to successfully train and validate supervised models. Finally, we demonstrate that our approach significantly reduces vulnerability of such models to model inversion attacks.**

Index Terms: Machine learning, Privacy, Neural nets

1. Introduction

The rise of data analytics and machine learning (ML) presents countless opportunities for companies, governments and individuals to benefit from the accumulated data. At the same time, their ability to capture fine levels of detail potentially compromises privacy of data providers. Recent research [1] suggests that even in a black-box setting it is possible to detect the presence of individual examples in the training set or recover certain features of these examples.

Among methods that tackle privacy issues of ML is the recent concept of *federated learning* (FL) [2]. In the FL setting, a central entity (*server*) trains a model without actually collecting user data. Instead, users (*clients*) update models locally, and the *server* aggregates these models. One popular approach is the federated averaging, `FedAvg` [2], where *clients* do on-device gradient descent using their data, then send these updates to the *server* where they get averaged. Privacy can be enhanced by using secure multi-party computation (MPC) to disallow the server access individual updates before averaging.

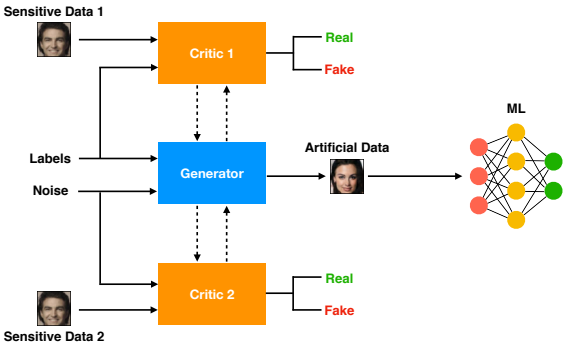

Figure 1: Architecture of our solution for two clients. Sensitive data is used to train a GAN (local critic and federated generator) to produce a private artificial dataset.

Despite many advantages, federated learning does have a number of challenges. First, the result of FL is a single trained model (therefore, we will refer to it as a *model release* method), which does not provide much flexibility in the future. For instance, it would significantly reduce

possibilities for further aggregation from different sources, e.g. different hospitals trying to combine federated models trained on their patients data. Second, this solution requires data to be labelled at the source, which is not always possible, because user may be unqualified to label their data or unwilling to do so. A good example is again a medical application where users are unqualified to diagnose themselves but at the same time would want to keep their condition private. Third, it does not offer provable privacy guarantees and *is vulnerable to attacks* like model inversion [1]. Some papers propose to augment FL with *differential privacy (DP)* [3] to alleviate this issue and provide rigorous theoretical guarantees. While these approaches perform well in ML tasks and provide theoretical privacy guarantees, they are often restrictive (e.g. many DP methods for ML assume, implicitly or explicitly, access to public data of similar nature or abundant amounts of data, which is not always realistic).

We address these problems by combining the strengths of federated learning and recent advancements in generative models to perform privacy-preserving *data release*. The main idea of our approach, named `FedGP`, for *federated generative privacy*, is to train generative adversarial networks (GANs) [4] on clients to produce artificial data that can replace clients real data. These generated samples can then be used for analytics and training machine learning models. Since some clients may have insufficient data to train a GAN locally, we instead train a federated GAN model. This way, user data always remain on their devices. Moreover, the federated GAN will produce samples from the common cross-user distribution and not from a single user, which adds to overall privacy. Figure 1 depicts the schematics of our approach.

This approach allows releasing entire datasets, which has many immediate advantages compared to *model release*. First, the released data could be used to train any ML model (we refer to it as *downstream task* or *downstream model*) without additional assumptions. Second, data from different sources could be easily pooled, allowing for hierarchical aggregation and building stronger models. Third, labelling and verification can be done later down the pipeline, relieving some trust and expertise requirements on users. Fourth, released data could be traded on data markets (https://www.datamakespossible.com/value-of-data-2018/dawn-of-data-marketplace), where anonymisation and protection of sensitive information is one of the biggest obstacles. Finally, data publishing would facilitate transparency and reproducibility of research.

To evaluate potential privacy risks, we use our *post hoc* privacy analysis framework [5] designed for private data release using GANs. Its key idea is to estimate KL divergence between pairs of synthetic data distributions produced by GANs with one-point difference in the original dataset.

Our contributions are the following:

- we extend our approach for private data release to the federated setting, broadening its applicability and enhancing privacy;
- we modify the federated learning protocol to allow a range of benefits mentioned above;
- we demonstrate that downstream models trained on artificial data achieve high accuracy while maintaining good average-case privacy and resilience to model inversion attacks.

2. Related Work

In recent years, as machine learning applications become a commonplace, several important vulnerabilities and corresponding attacks on ML models have been discovered. Model inversion [1] is based on observing output probabilities of the target model for a given class and performing gradient descent to obtain the training data reconstruction. Note that this attack can be performed in a black-box setting, without access to internal model parameters.

Most of the ML-specific literature in the area concentrates on privacy-preserving model release. Major solutions use differentially private training and have also been extended to federated learning [3].

A more recent line of research focuses on private data release and providing privacy via generating synthetic data [6]. In this scenario, DP is hard to guarantee, and thus, such models either relax the privacy notion or remain limited to simpler data. A recent approach explored by the community is training GANs with DP [6]. However, it proved extremely difficult to stabilise training with the necessary amount of noise,

which makes these methods inapplicable to more complex datasets without resorting to unrealistic (at least for some areas) assumptions, like access to public data from the same distribution. Finally, a hybrid model/data release solution by Fioretto and Van Hentenryck [7] employs decision trees and guarantees stronger ε-differential privacy, although like other data release approaches, it is less suitable for complex and continuous data like images.

3. Preliminaries

This section provides necessary definitions and background. Let us commence with approximate differential privacy.

Definition 1. *A randomised function (mechanism) $\mathcal{M} : \mathcal{D} \to \mathcal{R}$ with domain $\mathcal{D}$ and range $\mathcal{R}$ satisfies (ε, δ)-differential privacy if for any two adjacent inputs $d, d' \in \mathcal{D}$ and for any outcome $o \in \mathcal{R}$ the following holds:*

$$\Pr\left[\mathcal{M}(d) = o\right] \leq e^{\varepsilon} \Pr\left[\mathcal{M}(d') = o\right] + \delta. \quad (1)$$

Definition 2. *Privacy loss of a randomised mechanism $\mathcal{M} : \mathcal{D} \to \mathcal{R}$ for inputs $d, d' \in \mathcal{D}$ and outcome $o \in \mathcal{R}$ takes the following form:*

$$L_{(\mathcal{M}(d)\|\mathcal{M}(d'))} = \log \frac{\Pr\left[\mathcal{M}(d) = o\right]}{\Pr\left[\mathcal{M}(d') = o\right]}. \quad (2)$$

When clear from the context, we omit the subscript and simply denote privacy loss by L.

Definition 3. *The Kullback–Leibler (KL) divergence between two continuous probability distributions P and Q with corresponding densities p, q is given by:*

$$D_{KL}(P\|Q) = \mathbb{E}_{x\sim p(x)}\left[\log \frac{p(x)}{q(x)}\right]. \quad (3)$$

Combining Definitions 2 and 3, we see that the expectation of the privacy loss random variable $\mathbb{E}[L]$ is actually the KL divergence between the distributions of $\mathcal{M}(d)$ and $\mathcal{M}(d')$.

Finally, we use the Bayesian perspective on estimating mean from the data to get sharper bounds on expected privacy loss compared to the original work [5].

Proposition 1. *Let $[l_1, l_2, \ldots, l_m]$ be a random vector drawn from the distribution $p(L)$ with some common mean and variance, and let $\overline{L}$ and S be the sample mean and the sample standard deviation of the random variable L. Then,*

$$\Pr\left(\mathbb{E}[L] > \overline{L} + \frac{F^{-1}_{m-1}(1-\gamma)}{\sqrt{m-1}} S\right) \leq \gamma, \quad (4)$$

where $F^{-1}_{m-1}(1-\gamma)$ is the inverse CDF of the Student's t-distribution with $m-1$ degrees of freedom at $1-\gamma$.

The proof of this proposition can be obtained in the following way. Assuming the existence of the common mean and variance, we can use the maximum entropy principle for the likelihood function of these samples to ensure the highest uncertainty, and thus, conservativeness of the estimate. Combined with a flat prior, this likelihood function gives us the marginal distribution of the true mean $\mathbb{E}[L]$, and we observe that the random variable $\frac{\mathbb{E}[L]-\overline{L}}{S/\sqrt{m-1}}$ follows the Student's t-distribution with $m-1$ degrees of freedom [8]. We can then use the inverse of the Student's t CDF to arrive to Proposition 1.

4. Federated Generative Privacy

In order to keep participants data private while still maintaining flexibility in downstream tasks, our algorithm produces a federated generative model. This model can output artificial data, not belonging to any real user in particular, but coming from the common cross-user data distribution.

Let $\{u_1, u_2, \ldots, u_n\}$ be a set of *clients* holding private datasets $\{d_1, d_2, \ldots, d_n\}$. Before starting the training protocol, the *server* is providing each *client* with generator G_i^0 and critic C_i^0 models, and *clients* initialise their models randomly. Like in a normal FL setting, the training process afterwards consists of communication rounds. In each round t, *clients* update their respective models performing one or more passes through their data and submit generator updates $\triangle G_i^t$ to the *server* through MPC while keeping C_i^t private. In the beginning of the next round, the *server* provides an updated common generator G^t to all *clients*.

This approach has important advantages:

- Data do not physically leave user devices.
- Only generators (that do not come directly into contact with data) are shared, and critics remain private.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/MIS.2020.2993966, IEEE Intelligent Systems

- Using artificial data in downstream tasks adds another layer of protection and limits information leakage to artificial samples.

What remains to assess is how much information would an attacker gain about original data. We do so by employing a notion introduced in an earlier work [5] that we name *Differential Average-Case Privacy (DAP)*.

It is important to clarify why we do not use the standard DP to provide stronger theoretical guarantees: we found it extremely difficult to train GANs with the amount of noise required for meaningful DP guarantees. Despite a number of attempts (e.g. [6]), we are not aware of any technically sound solution that would generalise beyond simple datasets.

4.1. Differential Average-Case Privacy

Our framework builds upon ideas of *empirical DP* (EDP) and *on-average KL privacy* (for more details on related literature, we refer the reader to [5]). The first can be viewed as a measure of sensitivity on posterior distributions of outcomes (in our case, generated data distributions), while the second relaxes DP notion to the average case.

More specifically, we say the mechanism $\mathcal{M}$ is (μ, γ)-DAP if for two neighbouring datasets D, D', where data come from an observed distribution, it holds that

$$\Pr(\mathbb{E}[|L|] > \mu) \leq \gamma, \tag{5}$$

where L is the privacy loss (see Definition 2). Compare with a slightly rewritten definition of (ε, δ)-DP (which implies Definition 1):

$$\Pr(L > \varepsilon) \leq \delta. \tag{6}$$

For the sake of example, let each data point in D, D' represent a single user. Then, $(0.01, 0.001)$-DAP could be interpreted as follows: with probability 0.999, a typical user submitting their data will change outcome probabilities of the private algorithm on average by 1% (because $e^{0.01} \approx 1.01$.).

4.2. Generative Differential Average-Case Privacy

In the case of generative models, and in particular GANs, we don't have access to exact posterior distributions, a straightforward EDP procedure in our scenario would be the following: *(1)* train GAN on the original dataset D; *(2)* remove a random sample from D; *(3)* re-train GAN on the updated set; *(4)* estimate probabilities of all outcomes and the maximum privacy loss value; *(5)* repeat *(1)–(4)* sufficiently many times to approximate ε, δ.

If the generative model is simple, this procedure can be used without modification. Otherwise, for models like GANs, it becomes prohibitively expensive due to repetitive re-training (steps *(1)–(3)*). Another obstacle is estimating the maximum privacy loss value (step *(4)*). To overcome these two issues, we propose the following.

First, to avoid re-training, we imitate the removal of examples directly on the generated set $\widetilde{D}$. We define a similarity metric $sim(x, y)$ between two data points x and y that reflects important characteristics of data (see Section 5 for details). For every randomly selected real example i, we remove k nearest artificial neighbours and obtain $\widetilde{D}^{-i}$. Our intuition behind this operation is the following. Removing a real example would result in a lower probability density in the corresponding region of space. If this change is picked up by a GAN, which we assume is properly trained (e.g. there is no mode collapse), the density of this region in the generated examples space should also decrease. The number of neighbours k is defined by the ratio of artificial and real examples, to keep density normalised.

Second, we relax the worst-case privacy loss bound in step *(4)* by the expected-case bound, in the same manner as on-average KL privacy. This relaxation allows us to use a high-dimensional KL divergence estimator [9] to obtain the expected privacy loss for every pair of adjacent datasets $\widetilde{D}$ and $\widetilde{D}^{-i}$ (we denote it by $\mathcal{D}_{KL}^{-i}$, where $i = 1..m$). There are two major advantages of this estimator: it converges almost surely to the true value of KL divergence (see Definition 3); and it does not require intermediate density estimates to converge to the true probability measures. Also since this estimator uses nearest neighbours to approximate KL divergence, our heuristic described above is naturally linked to the estimation method.

Finally, having obtained sufficiently many sample pairs $(\widetilde{D}, \widetilde{D}^{-i})$, we use Proposition 1 to determine DAP parameters μ and γ. More specifically, we fix γ at the desired level (gener-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/MIS.2020.2993966, IEEE Intelligent Systems

Table 1: Accuracy of student models trained on artificial samples of FedGP compared to non-private centralised baseline and CentGP. In parenthesis we specify the average number of data points per client.

Setting	Dataset	Baseline	MD-GAN	CentGP	FedGP
i.i.d.	MNIST (500)	98.10%	64.30%	97.35%	79.45%
	MNIST (1000)	98.55%	93.46%	97.39%	93.38%
	MNIST (2000)	98.92%	97.47%	97.41%	96.23%
non-i.i.d.	MNIST (500)	97.31%	79.23%		83.26%
	MNIST (1000)	98.78%	91.90%	—	95.89%
	MNIST (2000)	98.76%	95.18%		96.88%

ally, inversely proportional to the number of data points), and then compute

$$\mu = \overline{L} + \frac{F_{m-1}^{-1}(1-\gamma)}{\sqrt{m-1}} S, \tag{7}$$

where $\overline{L}$ and S are the sample mean and the sample standard deviation of the sequence $\{\mathcal{D}_{KL}^{-i}\}$. This improvement over the original DAP gets a much tighter estimate of expected privacy loss.

4.3. Limitations

Our approach has a number of limitations that should be taken into consideration.

First of all, existing limitations of GANs (or generative models in general), such as training instability or mode collapse, will apply to this method. Hence, at the current state of the field, our approach may be difficult to adapt to inputs other than image data. Yet, there is still a number of privacy-sensitive applications, e.g. medical imaging or facial analysis, that could benefit from our technique. And as generative methods progress, new uses will be possible.

Second, since critics remain private and do not leave user devices their performance can be hampered by a small number of training examples. Nevertheless, we observe that even in the setting where some users have smaller datasets overall discriminative ability of all critics is sufficient to train good generators.

Lastly, our empirical privacy guarantee is not as strong as the traditional DP (e.g. it only estimates the average-case loss, and not the worst-case). However, due to the lack of DP-achieving training methods for GANs it is still beneficial to have an estimate of expected privacy loss rather than not having any guarantee.

5. Evaluation

We evaluate two major aspects of our method. First, we show that training ML models on data created by the common generator achieves high accuracy on MNIST (Section 5.1). Second, we estimate expected privacy loss of the federated GAN and evaluate the effectiveness of artificial data against model inversion attacks on CelebA face attributes (Section 5.2).

We choose two commonly used image datasets, MNIST (http://yann.lecun.com/exdb/mnist/) and CelebA (http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html). MNIST is a handwritten digit recognition dataset consisting of 60000 training examples and 10000 test examples, each example is a 28x28 size greyscale image. CelebA is a facial attributes dataset with 202599 images, each of which we crop to 128x128 and then downscale to 48x48.

In our experiments, we use Python and Pytorch framework. For implementation details of GANs and privacy evaluation, please refer to [5]. To train the federated generator we use FedAvg algorithm [2]. As a *sim* function introduced in Section 4.2 we use the distance between InceptionV3 feature vectors.

5.1. Learning Performance

First, we evaluate the generalisation ability of the student model trained on artificial data. The experiments are set up as follows:

1) Train the federated generative model (*teacher*) on the original distributed data.
2) Generate an artificial dataset and use it to train ML models (*students*).
3) Evaluate students on a held-out test set.

We compare learning performance with the baseline centralised model trained on original data, as well as the same model trained on artificial samples obtained from the centrally trained GAN (`CentGP`) and from MD-GAN [10]. The latter is another distributed GAN approach, dif-

Table 2: Average-case privacy parameters: expected privacy loss bounds μ_C and μ_F (for centralised and federated solutions), and probability γ of exceeding it. A typical ε of DP in this setting is > 2.

Setting	Dataset	μ_C	μ_F	γ
i.i.d.	MNIST (500)	0.0101	0.0117	10^{-15}
	MNIST (1000)	0.0046	0.0069	
	MNIST (2000)	0.0015	0.0021	
	CelebA	0.0009	0.0009	
non-i.i.d.	MNIST (500)	–	0.0090	10^{-15}
	MNIST (1000)	–	0.0044	
	MNIST (2000)	–	0.0020	

fering from our federated GAN by the fact that critics are randomly exchanged between clients in a peer-to-peer fashion.

Since critics stay private in FedGP and only access data of a single user, the size of each individual dataset has significant effect. Therefore, in our experiment we vary sizes of user datasets and observe its influence on training. In each experiment, we specify an average number of points per user, while the actual number is drawn from the uniform distribution with this mean, with some clients getting as few as 100 data points.

We also study two settings: i.i.d. and non-i.i.d data. In the first setting, distribution of classes for each client is identical to the overall distribution. In the second, every client gets samples of 2 random classes, imitating the situation when a single user observes only a part of overall data distribution.

Details of the experiment can be found in Table 1. We observe that training on artificial data from the federated GAN allows to achieve 96.9% accuracy on MNIST with the baseline of 98.8%. We can also see how accuracy grows with the average user dataset size. A less expected observation is that non-i.i.d. setting is actually beneficial for FedGP. A possible reason is that training critics with little data becomes easier when this data is less diverse (i.e. the number of different classes is smaller).

We find that the performance of MD-GAN is similar to FedGP in the i.i.d. case and is slightly behind in the non-i.i.d. case. Therefore, we believe that the additional privacy leakage and the extra communication complexity of MD-GAN associated with the critics exchange are not justified in the examined setting. Comparing to the centralised generative privacy model CentGP, we can see that FedGP is more affected by sharding of data on user devices than by overall data size, suggesting that further research in training federated generative models is necessary.

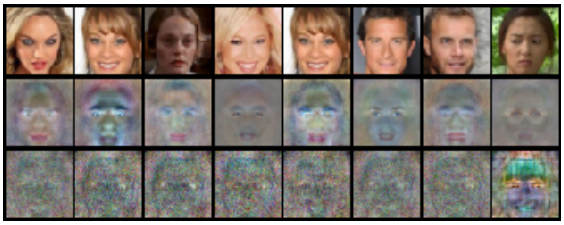

Figure 2: Results of the model inversion attack. Top to bottom: real target images, reconstructions from the non-private model, reconstructions from the model trained by FedGP.

Table 3: Face detection and recognition rates (pairs with distances below 0.99) for images recovered by model inversion from the non-private baseline and the FedGP-trained model.

	Baseline	FedGP
Detection	25.5%	1.2%
Recognition	2.8%	0.1%

5.2. Privacy Analysis

Using the privacy estimation framework (see Sections 4.1 and 4.2), we fix the probability γ of exceeding the expected privacy loss bound μ in all experiments to 10^{-15} and compute the corresponding μ. Table 2 summarises the bounds we obtain. As anticipated, the privacy guarantee improves with the growing number of data points, because the influence of each individual example diminishes. Moreover, the average privacy loss μ, expectedly, is significantly smaller than the typical worst-case DP loss ε in similar settings (between 2 to 10, or even larger). To put it in perspective, the average change in outcome probabilities estimated by DAP is $\sim 1\%$ even in more difficult settings, while the state-of-the-art DP method would place the worst-case change at $> 100\%$ or even $> 1000\%$ without giving much information about a typical case. Compared to the centralised solution (μ_C), the federated version may have slightly weaker privacy guarantees, probably because of the higher degree of overfitting for critics. But this difference diminishes with growing data size, and for CelebA μ_F actually gets smaller than μ_C.

On top of estimating expected privacy loss bounds, we test FedGP's resistance to the *model*

inversion attack [1]. More specifically, we run the attack on two student models: trained on original data samples and on artificial samples correspondingly. Note that we also experimented with another well-known attack on machine learning models, the membership inference. However, we did not include it in the final evaluation, because of the poor attacker's performance in our setting (nearly random guess accuracy for given datasets and models even on the non-private baseline). Moreover, we only consider *passive adversaries* and we leave evaluation with active adversaries, for future work.

In order to run the attack, we train a student model (a simple MLP with two hidden layers of 1000 and 300 neurons) to similar accuracy levels in two settings: the real data and the artificial data generated by `FedGP`. As facial recognition is a more privacy-sensitive application, and provides a better visualisation of the attack, we pick the CelebA dataset for this experiment.

We analyse real and reconstructed image pairs using OpenFace [11] (see Table 3). It confirms our theory that artificial samples would shield real data in case of the downstream model attack. In the images reconstructed from a non-private model, faces were detected 25.5% of the time and recognised in 2.8% of cases. For our method, detection succeeded only in 1.2% of faces and the recognition rate was 0.1%, well within the state-of-the-art error margin for face recognition.

Figure 2 shows results of the model inversion attack. The top row presents the real target images. The following rows depict reconstructed images from the non-private model and the model trained on the federated GAN samples. One can observe a clear information loss in reconstructed images going from the non-private to the `FedGP`-trained model. Despite failing to conceal general shapes in training images (i.e. faces), our method seems to achieve a trade-off, hiding most of the specific features, while the non-private model reveals important facial features, such as skin and hair colour, expression, etc. The obtained reconstructions are either very noisy or converge to some average feature-less faces.

6. Conclusions

We study the intersection of federated learning and private data release using GANs. Combined these methods enable important advantages and applications for both fields, such as higher flexibility, reduced trust and expertise requirements on users, hierarchical data pooling, and data trading.

The choice of GANs as a generative model ensures scalability and makes the technique suitable for real-world data with complex structure. In our experiments, we show that student models trained on artificial data can achieve high accuracy on classification tasks. Moreover, models can also be validated on artificial data. Importantly, unlike many prior approaches, our method does not assume access to similar publicly available data.

We estimate and bound the expected privacy loss of an average client by using differential average-case privacy thus enhancing privacy of traditional federated learning. We find that, in most scenarios, the presence or absence of a single data point would not change the outcome probabilities by more than 1% on average. Additionally, we evaluate the provided protection by running the model inversion attack and showing that training with the federated GAN reduces information leakage (e.g. face detection in recovered images drops from 25.5% to 1.2%).

REFERENCES

1. M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in *Proc. ACM CCS*. ACM, 2015, pp. 1322–1333.
2. H. B. McMahan, E. Moore, D. Ramage, S. Hampson *et al.*, "Communication-efficient learning of deep networks from decentralized data," *arXiv preprint arXiv:1602.05629*, 2016.
3. H. B. McMahan, D. Ramage, K. Talwar, and L. Zhang, "Learning differentially private recurrent language models," *arXiv preprint arXiv:1710.06963*, 2017.
4. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2014, pp. 2672–2680.
5. A. Triastcyn and B. Faltings, "Generating artificial data for private deep learning," in *Proc. PAL, AAAI Spring Symposium Series*, 2019.
6. B. K. Beaulieu-Jones, Z. S. Wu, C. Williams, and C. S. Greene, "Privacy-preserving generative deep neural networks support clinical data sharing," *bioRxiv*, p. 159756, 2017.

7. F. Fioretto and P. Van Hentenryck, "Privacy-preserving federated data sharing," in *Proc. AAMAS 2019*, 2019, pp. 638–646.
8. T. E. Oliphant, "A bayesian perspective on estimating mean, variance, and standard-deviation from data," 2006.
9. F. Pérez-Cruz, "Kullback-leibler divergence estimation of continuous distributions," in *Information Theory, 2008. ISIT 2008. IEEE International Symposium on*. IEEE, 2008, pp. 1666–1670.
10. C. Hardy, E. Le Merrer, and B. Sericola, "Md-gan: Multi-discriminator generative adversarial networks for distributed datasets," in *2019 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. IEEE, 2019, pp. 866–877.
11. B. Amos, B. Ludwiczuk, M. Satyanarayanan *et al.*, "Openface: A general-purpose face recognition library with mobile applications," 2016.

Aleksei Triastcyn is a graduate student at the Artificial Intelligence Laboratory at EPFL, Switzerland. His research is focused on privacy-preserving machine learning. Contact him at aleksei.triastcyn@epfl.ch.

Boi Faltings is a full professor and director of the Artificial Intelligence Laboratory at EPFL, Switzerland. He received the PhD degree from the University of Illinois. His research interests include game-theoretic information elicitation, multi-agent systems, recommender systems, and intelligent user interfaces. Contact him at boi.faltings@epfl.ch.