Abstract

In this paper, we propose generating artificial data that retain statistical properties of real data as the means of providing privacy with respect to the original dataset. We use generative adversarial network to draw privacy-preserving artificial data samples and derive an empirical method to assess the risk of information disclosure in a differential-privacy-like way. Our experiments show that we are able to generate artificial data of high quality and successfully train and validate machine learning models on this data while limiting potential privacy loss.

1 Introduction

Following recent advancements in deep learning, more and more people and companies get interested in putting their data in use and employ machine learning (ML) to generate a wide range of benefits that span financial, social, medical, security, and other aspects. At the same time, however, such models are able to capture a fine level of detail in training data, potentially compromising privacy of individuals whose features sharply differ from others. Recent research by Fredrikson et al. (2015) suggests that even without access to internal model parameters, by using hill climbing on output probabilities of a neural network, it is possible to recover (up to a certain degree) individual examples (e.g. faces) from a training set.

The latter result is especially disturbing knowing that deep learning models are becoming an integral part of our lives, making its way to phones, smart watches, cars, and appliances. And since these models are often trained on customers’ data, such training set recovery techniques endanger privacy even without access to the manufacturer’s servers where these models are being trained.

One approach to this problem explored in recent literature is enforcing privacy during training (Abadi et al., 2016; Papernot et al., 2016, 2018; Bindschaedler et al., 2017). While these methods perform well in ML tasks and provide strong privacy guarantees, they are often restrictive. For example, evaluating a variety of models to pick the best one is complicated by the need of adjusting private training for each of them. Moreover, most of these methods assume (implicitly or explicitly) access to public data of similar nature, which may not be possible in areas like medicine.

In this work, we take an alternative approach. We study the task of publishing datasets in a privacy-preserving manner, such that any ML model could be trained on it without additional assumptions. In particular, we are interested in solving two problems. First, how to preserve high utility of data for ML algorithms while protecting sensitive information in the dataset. Second, how to quantify the risk of recovering private information from the published dataset, and thus, the trained ML model.

The main idea of our approach is to use generative adversarial networks (GANs) (Goodfellow et al., 2014) to create artificial datasets to be used in place of real data for training. This method has a number of advantages over the earlier work (Abadi et al., 2016; Papernot et al., 2016, 2018; Bindschaedler et al., 2017). First of all, our solution allows releasing entire datasets to any (untrusted)
ML service providers. Second, it achieves high accuracy without pre-training on similar public data. Third, it is more intuitive and flexible, e.g. it does not require a complex distributed architecture.

To evaluate potential privacy risks of the technique, we design a framework for ex post analysis of generated data. We use KL divergence estimation and Chebyshev’s inequality to find a statistical bound on expected privacy loss for a specific dataset in question.

Our contributions in this paper are the following:

- we propose a novel, yet simple, approach for private data release, and to the best of our knowledge, this is the first practical solution for complex real-world data;
- we introduce a new framework for statistical evaluation of potential privacy loss of the released data;
- we show that our method achieves a good privacy-accuracy trade-off in learning tasks.

The remainder of the paper is structured as follows. In Section 2 we give an overview of related work. Section 3 contains necessary background. In Section 4 we describe our approach and privacy evaluation framework, as well as discuss its limitations. Experimental results and implementation details are presented in Section 5, and Section 6 concludes the paper.

2 Related work

In order to protect privacy while still benefiting from the use of statistics and ML, many techniques for data anonymisation have been developed over the years, including \( k \)-anonymity (Sweeney, 2002), \( l \)-diversity (Machanavajjhala et al., 2007), \( t \)-closeness (Li et al., 2007), and differential privacy (DP) (Dwork, 2006). The latter has been recognised as a rigorous standard and is widely accepted by the research community. However, its generic formulation makes it hard to quantify potential privacy loss of the already trained model and in a data-dependent way. We adapt a notion of empirical DP (Abowd et al., 2013) to our setting in order to overcome this limitation.

A number of approaches has been proposed in the literature to tackle the task of privacy-preserving ML. One take on the problem is to distribute training and use disjoint datasets. For example, Shokri & Shmatikov (2015) propose to train a model in a distributed manner by communicating sanitised updates from participants to a central authority. Such a method, however, yields high privacy losses as pointed out by Abadi et al. (2016) and Papernot et al. (2016). An alternative technique suggested by Papernot et al. (2016), also uses disjoint training sets and builds an ensemble of independently trained teacher models to transfer knowledge to a student model by labelling public data. This result has been extended in (Papernot et al., 2018) to achieve state-of-the-art private (single-digit DP bounds) accuracy on a number of image classification tasks. A different approach is taken by Abadi et al. (2016). They suggest using differentially private stochastic gradient descent (DP-SGD) to train deep learning models in a private manner. This approach allows to achieve high accuracy while maintaining low DP bounds, but may also require pre-training on public data.

A more recent line of research focuses on providing privacy via generating synthetic data (Bindschaedler et al., 2017; Huang et al., 2017; Beaulieu-Jones et al., 2017). In this scenario, DP is hard to guarantee, and thus, such models either relax the DP requirements or remain limited to simple data. In Bindschaedler et al. (2017), the authors use a graphical probabilistic model to learn an underlying data distribution and transform real data points (seeds) into synthetic data points, which are then filtered by a privacy test based on a plausible deniability criterion. This procedure would be rather expensive for complex data, such as images. Huang et al. (2017) introduce the notion of generative adversarial privacy and use GANs to obfuscate real data points w.r.t. pre-defined private attributes, enabling privacy for more realistic datasets. Finally, a natural approach to try is training GANs using DP-SGD (Beaulieu-Jones et al., 2017; Xie et al., 2018; Zhang et al., 2018). However, it’s been proven extremely difficult to stabilise training with the necessary amount of noise, which scales as \( \sqrt{M} \) w.r.t. the number of model parameters \( M \).

Similarly, our approach employs GANs, but data is generated without real seeds or applying noise to gradients. Instead, we verify experimentally that out-of-the-box GAN samples can be sufficiently different from real data, and expected privacy loss is empirically bounded by single-digit numbers.
Figure 1: Architecture of our solution. Sensitive data is used to train GAN (consisting of critic and generator) to produce a private artificial dataset, which then can be used by any ML model.

3 Preliminaries

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

**Definition 1.** A randomised function (mechanism) \( M : D \rightarrow R \) with domain \( D \) and range \( R \) satisfies \((\varepsilon, \delta)\)-differential privacy if for any two adjacent inputs \( d, d' \in D \) and for any outcome \( o \in R \) the following holds:

\[
Pr[M(d) = o] \leq e^\varepsilon Pr[M(d') = o] + \delta. \tag{1}
\]

**Definition 2.** Privacy loss of a randomised mechanism \( M : D \rightarrow R \) for outcome \( o \in R \) is a random variable of the following form:

\[
L(M(d)||M(d')) = \log \frac{Pr[M(d) = o]}{Pr[M(d') = o]}. \tag{2}
\]

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

\[
D_{KL}(P||Q) = \int_{-\infty}^{+\infty} p(x) \log \frac{p(x)}{q(x)} dx. \tag{3}
\]

Note that KL divergence between the distributions of \( M(d) \) and \( M(d') \) is nothing but the expectation of the privacy loss random variable \( E[L(M(d)||M(d'))] \).

Finally, we will be using Chebyshev’s inequality to obtain tail bounds. In particular, as we expect the distribution to be asymmetric, we use a version of Chebyshev’s inequality with semi-variances \( \text{Berck} \ & \text{Hihn} \ [1982] \) to get a sharper bound:

\[
Pr(x \geq E[x] + k\sigma) \leq \frac{1}{k^2} \frac{\sigma^2}{\sigma^2}, \tag{4}
\]

where \( \sigma^2 = \int_{E[x]}^{+\infty} p(x)(x - E[x])^2 dx \) is the upper semi-variance.

4 Our Approach

In this section, we describe our solution, its further improvements, and provide details of the privacy evaluation framework. We then discuss limitations of the method. The general workflow of our solution is depicted in Figure 1.
The main idea of our approach is to use artificial data instead of real data for learning purposes. The intuition behind it is the following. Since it is possible to recover training examples from ML models (Fredrikson et al. 2015), one needs to limit the exposure of real data during training. While this can be achieved by training with differential privacy (e.g. using DP-SGD), it has been shown that certain attacks can still be successful due to leveraging loose DP bounds (Hitaj et al. 2017). Removing real data from the training process altogether would allow to add another layer of protection and would limit the information leakage to artificial samples. What remains is to show that synthetic data generated by GAN is sufficiently different from real sensitive data.

### 4.1 Differentially Private Critic

Despite the fact that the generator does not have access to real data in the training process, one cannot guarantee that generated samples will not repeat the input. To alleviate this problem, we propose to enforce differential privacy on the output of the discriminator (critic). This is done by clipping the $L^2$ norm of the second-to-last layer and adding Gaussian noise. To be more specific, activations $a(x)$ of the second-to-last layer become $\tilde{a}(x) = a(x) / \max(\|a(x)\|_2, 1) + \mathcal{N}(0; \sigma^2)$. We refer to this version of the critic as DP critic.

Note that if the chosen GAN loss function was directly differentiable w.r.t. generator output, i.e. if critic could be treated as a black box, this modification would enforce the same DP guarantees on generator parameters, and consequently, all generated samples. Unfortunately, this is not the case for practically all existing versions of GANs, including WGAN-GP (Gulrajani et al. 2017) used in our experiments.

As our evaluation shows, this modification has a number of advantages. First, it improves diversity of samples and decreases similarity with real data. Second, it allows to prolong training, and hence, obtain higher quality samples. Finally, in our experiments, it significantly improves the ability of GANs to generate samples conditionally.

### 4.2 Privacy Evaluation Framework

As we pointed out, achieving differential privacy in GANs is a difficult task and leads to significant loss of quality due to excessive amount of noise needed to ensure small $\epsilon, \delta$. Furthermore, performance of DP-SGD will degrade substantially for large networks, as the added noise scales as $\sqrt{M}$ w.r.t. the number of model parameters $M$. We take an alternative approach by evaluating privacy ex post.

Our framework builds upon ideas of empirical DP (EDP) outlined in Abowd et al. (2013) for Bayesian Linear Mixed Models, and then extended to Bayesian Generalised Linear Mixed Models in Schneider & Abowd (2015). Relation between EDP and DP was investigated by Charest & Hou (2017), and the authors concluded that it can be viewed as a measure of sensitivity on posterior distributions of outcomes (in our case, generated data distributions).

As we don’t have access to exact posterior distributions, a straightforward 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) to obtain a sufficient number of samples, and then use Chebyshev’s inequality to approximate $\epsilon, \delta$.

This procedure, however, is computationally expensive, because it requires re-training GAN thousands of times (steps (1)–(3)). Another obstacle is estimating the maximum privacy loss value (step (4)). While it can be done using density estimators, the quality of such estimates in high-dimensional spaces would be very low. To overcome these two issues, we propose the following.

First, let us optimise steps (1)–(3). As re-training GANs is expensive, we imitate the removal of examples on the generated set $\tilde{D}$ directly. In order to do this, we define a similarity metric $\text{sim}\,(x,y)$ between two data points $x$ and $y$ that reflects distinguishable characteristics of data (see Section 5 for the specific metric used). For every randomly selected real example $i$, we remove $k$ nearest (in terms of $\text{sim}$-value) artificial neighbours to simulate the absence of this example in the training set and obtain $\tilde{D}^{-i}$.

It is easy to see that increasing $k$ would lead to a larger difference between the two artificial distributions, and therefore, higher privacy loss estimate. The exact number of points influenced by
a specific example is unknown but, considering that GANs are designed to duplicate the real data distribution, we heuristically choose $k$ by computing the KL divergence between the real ($D$) and artificial ($\tilde{D}$) data distributions and then assuming that all the difference comes from one point.

Moving on to the second issue, we propose to relax the worst-case privacy loss bound in step (4) by the expected-case bound. This relaxation allows us to use a high-dimensional KL divergence estimator (Pérez-Cruz, 2008) to obtain the expected privacy loss for every pair of adjacent datasets ($\tilde{D}$ and $\tilde{D}^{-i}$). There are two major advantages of this estimator: it converges almost surely to the true value of KL divergence; and it does not require intermediate density estimates to converge to the true probability measures.

Finally, after obtaining sufficiently many samples of expected privacy loss for different pairs ($\tilde{D}, \tilde{D}^{-i}$), we use Chebyshev’s inequality (see Eq. 4) to bound the probability $\gamma = \Pr(\mathbb{E}[L(M(D)||M(D'))] \geq \mu)$ of it exceeding a predefined threshold $\mu$.

### 4.3 Limitations

First and foremost limitation of our approach is the lack of strict differential privacy guarantees. Although we provide bounds on expected privacy loss based on ex post analysis of the generated dataset, it is not equivalent to the traditional formulation of DP (e.g. it only concerns a given dataset).

Second, our privacy evaluation framework itself could be improved in a number of ways. For instance, providing worst-case privacy loss bounds would be largely beneficial. Furthermore, simulating the removal of examples from the training set currently depends on heuristics and the chosen similarity metric, which may not lead to representative samples.

Lastly, all existing limitations of GANs, such as training instability or mode collapse, will apply to this method.

### 5 Evaluation

In this section, we describe the experimental setup and implementation, and evaluate our method on MNIST (LeCun et al., 1998), SVHN (Netzer et al., 2011), and CelebA (Liu et al., 2015) datasets.

#### 5.1 Experimental setup

We evaluate our method in four major ways. First, we show that not only is it feasible to train ML models purely on generated data, but it is also possible to achieve high learning performance (Section 5.3). Second, we demonstrate an even stronger result: generated data can be used as a validation set for tuning model hyper-parameters (Section 5.4). Third, we report Fréchet Inception Distance (FID) (Heusel et al., 2017) between real and generated datasets to highlight additional advantages of the DP critic (Section 5.5). Finally, we evaluate privacy of artificial data by computing empirical bounds on expected privacy loss (Section 5.6).

Learning and validation performance experiments are set up as follows:

1. Train a generative model (teacher) on the original dataset using only the training split.
2. Generate an artificial dataset by the obtained model and use it to train ML models (students).
3. Evaluate students on a held-out test set.

Note that there is no dependency between teacher and student models. Moreover, student models are not constrained to neural networks and can be implemented as any type of machine learning algorithm.

We choose three commonly used image datasets for our experiments: MNIST, SVHN, and CelebA. MNIST is a handwritten digit recognition dataset consisting of 60'000 training examples and 10'000 test examples, each example is a 28x28 size greyscale image. SVHN is also a digit recognition task, with 73'257 images for training and 26'032 for testing. The examples are coloured 32x32 pixel images of house numbers from Google Street View. CelebA is a facial attributes dataset with 202'599 images, each of which we crop to 128x128 and then downscale to 48x48.
Table 1: Accuracy of student models for non-private baseline, PATE (Papernot et al. 2016), and our method.

| Dataset | Non-Private Baseline | PATE  | Our approach |
|---------|----------------------|-------|--------------|
| MNIST   | 99.2%                | 98.0% | 98.3%        |
| SVHN    | 92.8%                | 82.7% | 87.7%        |

5.2 Implementation details

For our experiments, we use Python and PyTorch framework.[1] We implemented (with some modifications) an improved Wasserstein GAN (WGAN-GP) by Gulrajani et al. (2017). More specifically, the critic consists of four convolutional layers with SELU activations (Klambauer et al., 2017) followed by a fully connected linear layer which outputs a $d$-dimensional feature vector ($d = 64$). For the DP critic, we additionally clip the $L_2$-norm of this vector to $C = 1$ and add Gaussian noise with $\sigma = 1.5$ (we refer to it as DP layer). Finally, it is passed through a linear classification layer. The generator starts with a fully connected linear layer that transforms noise and labels into a 4096-dimensional feature vector which is then passed through a SELU activation and three deconvolution layers with SELU activations. The output of the third deconvolution layer is downsampled by max pooling and normalised with a tanh activation function.

Similarly to the original paper, we use a classical WGAN value function with the gradient penalty that enforces Lipschitz constraint on a critic. We also set the penalty parameter $\lambda = 10$ and the number of critic iterations $n_{\text{critic}} = 5$. Furthermore, we modify the architecture to allow for conditioning WGAN on class labels. Binarised labels are appended to the input of the generator and to the linear layer of the critic after convolutions. Therefore, the generator can be used to create labelled datasets for supervised learning.

Both networks are trained using Adam optimiser (Kingma & Ba, 2015) with parameters similar to (Gulrajani et al., 2017): learning rate set to 0.0001, $\beta_1 = 0$, $\beta_2 = 0.9$, and a batch size of 64.

The student network is constructed of two convolutional layers with ReLU activations, batch normalisation and max pooling, followed by two fully connected layers with ReLU, and a softmax output layer. It is worth mentioning that this network does not achieve state-of-the-art performance on the used datasets, but we are primarily interested in evaluating the performance drop compared to a non-private model rather than getting the best test score.

To evaluate privacy we implemented the procedure described in Section 4. In particular, building upon recent ideas in image qualitative evaluation, such as FID and Inception Score, we compute image feature vectors by a pre-trained Inception V3 network (Szegedy et al., 2016), and use inverse distances between these features as $\text{sim}(\cdot, \cdot)$ function. We implemented the KL divergence estimator by Pérez-Cruz (2008), and used k-d trees (Bentley, 1975) for fast nearest neighbour searches.

5.3 Learning Performance

First, we evaluate the generalisation ability of a student model trained on artificial data. More specifically, we train a student model on generated data and report test classification accuracy on a held-out real set.

As mentioned above, most of the existing work on privacy-preserving ML assumes access to similar public data in one form or another (Abadi et al., 2016; Papernot et al., 2016; 2018; Zhang et al., 2018). We expect no public data availability, and this assumption limits the choice of techniques for comparison. The current state-of-the-art method for considered datasets (PATE by Papernot et al. (2018)) treats a part of the test set as unlabelled public data, which is not directly comparable to our method. To make the evaluation more appropriate, we pick the results of PATE corresponding to the least number of labelling queries for public data.

Table 1 shows test accuracy results for the non-private baseline model (trained on the real training set), PATE, and our method. We observe that artificial data allows us to achieve 98.3% accuracy on MNIST and 87.7% accuracy on SVHN, which is comparable or better than corresponding results of

[1] http://pytorch.org
Table 2: Empirical privacy parameters: expected privacy loss bound $\mu$ and probability $\gamma$ of exceeding it.

| Dataset | Method                | $\mu$  | $\gamma$  |
|---------|-----------------------|--------|-----------|
| MNIST   | WGAN-GP               | 5.80   |           |
|         | WGAN-GP (DP critic)   | 5.36   |           |
| SVHN    | WGAN-GP               | 13.16  | $10^{-5}$ |
|         | WGAN-GP (DP critic)   | 4.92   |           |
| CelebA  | WGAN-GP               | 6.27   |           |
|         | WGAN-GP (DP critic)   | 4.15   |           |

PATE. It is worth mentioning again, that our method, unlike PATE, does not provide strict differential privacy guarantees, but neither does it assume public data availability.

Additionally, we trained a simple logistic regression model on artificial MNIST samples, and obtained 91.69% accuracy, compared to 92.58% on the original data, which confirms that student models are not restricted to a specific type.

5.4 Validation Performance

In the previous section, we demonstrated that ML models trained on artificial data can generalise well enough and achieve high accuracy on unseen real data. However, there is another important aspect of training: choosing the right hyper-parameters. In scenarios where public data is not available, validation has to be done on private artificial data, and to the best of our knowledge, the question of using artificial data for validation has not been well covered in the machine learning literature.

We evaluate the validation power in the following way. While training a student model, we compute validation loss on a real held-out set and on an artificial held-out set. We then compute correlation between the two sequences. Figure 2 shows two pairs of validation loss curves, for MNIST and SVHN datasets. We observe that, indeed, artificial validation loss closely follows the real one, despite being generally lower and fluctuating more. Note that lower validation loss does not imply better test performance, but high correlation is important for hyper-parameter tuning. We ran experiments for a number of different learning rates, and correlation coefficients range from 0.7197 to 0.9972 for MNIST and from 0.8047 to 0.9810 for SVHN.

5.5 Generated Samples Quality

While the main purpose of this work is to evaluate and improve privacy of data generated by GANs, we observe that addition of DP layer in the critic has a beneficial side effect: improving image quality and diversity, as well as providing regularisation effect which allows for much longer stable training.
Figure 3 shows FID values for every 10-th epoch of training with and without DP layer. For both SVHN and CelebA, WGAN-GP with DP critic achieves better performance and converges more stably. At the same time, the quality of CelebA samples for vanilla WGAN-GP significantly degrades after 100 epochs indicating overfitting (note that for privacy evaluation we chose the epoch with the best FID score). Moreover, GANs with DP critics achieve better FID scores for given datasets than the best ones reported in Heusel et al. (2017).

5.6 Privacy Analysis

We evaluated potential privacy loss as described in Section 4. We fixed the probability \( \gamma \) of exceeding the expected privacy loss bound in all experiments to \( 10^{-5} \), and computed the corresponding bound for each dataset and two versions of GAN (vanilla WGAN-GP, and WGAN-GP with DP critic). Table 2 summarises our findings: (1) empirical bounds on privacy loss are on par with DP bounds for deep learning found in recent literature (Abadi et al., 2016; Papernot et al., 2018); (2) DP critic helps to bring down \( \mu \) values in all cases. It is worth noting, however, that our \( \mu \) should not be directly compared to \( \varepsilon \) of DP, since \( \mu \) bounds expected privacy loss while \( \varepsilon \) bounds maximum privacy loss.

In addition to quantitative analysis of privacy loss bounds, we perform visual inspection of generated examples and corresponding nearest neighbours in real data. Figure 4 depicts generated and real images from SVHN and CelebA datasets. Each real sample is chosen to be the most similar to the corresponding generated example.

We observe that, at first glance, generated images can be very close to real examples visually. However, differences are in details, which is normally more important for privacy preservation. For SVHN, digits usually differ either in shape, colour or surroundings. Moreover, a lot of pairs come from different classes. For CelebA, the pose and lighting may be similar, but such details as gender, skin colour, facial features are usually significantly different.

6 Conclusions

We investigate the problem of private data release for complex high-dimensional data. We employ generative adversarial networks to produce artificial privacy-preserving datasets. The choice of GANs as a generative model ensures scalability and makes the technique suitable for real-world data with complex structure. Unlike many prior approaches, our method does not assume access to similar publicly available data. In our experiments, we show that student models trained on artificial data can achieve high accuracy on MNIST and SVHN datasets. Moreover, models can also be validated on artificial data.

We propose a novel technique for evaluating privacy of released data by computing empirical bounds on expected privacy loss. We compute privacy bounds for samples from WGAN-GP on MNIST, SVHN, and CelebA datasets, and demonstrate that expected privacy loss is bounded by single-digit values.

Additionally, we introduce a simple modification to the critic: differential privacy layer. It ensures DP guarantees for the critic output by adding noise in a low-dimensional embedding space during forward pass. Not only does it improve privacy loss bounds, but it also acts as a regulariser improving stability of training, and quality and diversity of generated images.
Considering the rising importance of privacy research and the lack of good solutions for private data publishing, there is a lot of potential future work. In particular, a major direction of advancing current work would be achieving differential privacy guarantees for GANs while still preserving high utility of generated data. A step in another direction would be to improve the privacy evaluation framework, e.g. by bounding maximum privacy loss, or finding a more principled way of sampling from outcome distributions.

References

Abadi, Martín, Chu, Andy, Goodfellow, Ian, McMahan, H Brendan, Mironov, Ilya, Talwar, Kunal, & Zhang, Li. 2016. Deep learning with differential privacy. Pages 308–318 of: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. ACM.

Abowd, John M, Schneider, Matthew J, & Vilhuber, Lars. 2013. Differential privacy applications to Bayesian and linear mixed model estimation. Journal of Privacy and Confidentiality, 5(1), 4.

Beaulieu-Jones, Brett K, Wu, Zhiwei Steven, Williams, Chris, & Greene, Casey S. 2017. Privacy-preserving generative deep neural networks support clinical data sharing. bioRxiv, 159756.

Bentley, Jon Louis. 1975. Multidimensional binary search trees used for associative searching. Communications of the ACM, 18(9), 509–517.

Berc, Peter, & Hihn, Jairus M. 1982. Using the semivariance to estimate safety-first rules. American Journal of Agricultural Economics, 64(2), 298–300.

Bindschaedler, Vincent, Shokri, Reza, & Gunter, Carl A. 2017. Plausible Deniability for Privacy-Preserving Data Synthesis. Proceedings of the VLDB Endowment, 10(5).

Charest, Anne-Sophie, & Hou, Yiwei. 2017. On the Meaning and Limits of Empirical Differential Privacy. Journal of Privacy and Confidentiality, 7(3), 3.

Dwork, Cynthia. 2006. Differential Privacy. Pages 1–12 of: 33rd International Colloquium on Automata, Languages and Programming, part II (ICALP 2006), vol. 4052. Venice, Italy: Springer Verlag.

Fredrikson, Matt, Jha, Somesh, & Ristenpart, Thomas. 2015. Model inversion attacks that exploit confidence information and basic countermeasures. Pages 1322–1333 of: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. ACM.

Goodfellow, Ian, Pouget-Abadie, Jean, Mirza, Mehdi, Xu, Bing, Warde-Farley, David, Ozair, Sherjil, Courville, Aaron, & Bengio, Yoshua. 2014. Generative adversarial nets. Pages 2672–2680 of: Advances in Neural Information Processing Systems.

Gulrajani, Ishaan, Ahmed, Faruk, Arjovsky, Martin, Dumoulin, Vincent, & Courville, Aaron C. 2017. Improved training of wasserstein gans. Pages 5769–5779 of: Advances in Neural Information Processing Systems.

Heusel, Martin, Ramsauer, Hubert, Unterthiner, Thomas, Nessler, Bernhard, Klambauer, Günter, & Hochreiter, Sepp, 2017. GANs trained by a two time-scale update rule converge to a Nash equilibrium. arXiv preprint arXiv:1706.08500.

Hitaj, Briland, Ateniese, Giuseppe, & Pérez-Cruz, Fernando. 2017. Deep models under the GAN: information leakage from collaborative deep learning. Pages 603–618 of: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. ACM.

Huang, Chong, Kairouz, Peter, Chen, Xiao, Sankar, Lalitha, & Rajagopal, Ram. 2017. Context-aware generative adversarial privacy. Entropy, 19(12), 656.

Kingma, Diederik, & Ba, Jimmy. 2015. Adam: A method for stochastic optimization. In: Proceedings of the 3rd International Conference for Learning Representations.

Klambauer, Günter, Unterthiner, Thomas, Mayr, Andreas, & Hochreiter, Sepp. 2017. Self-normalizing neural networks. Pages 972–981 of: Advances in Neural Information Processing Systems.
LeCun, Yann, Bottou, Léon, Bengio, Yoshua, & Haffner, Patrick. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324.

Li, Ninghui, Li, Tiancheng, & Venkatasubramanian, Suresh. 2007. t-closeness: Privacy beyond k-anonymity and l-diversity. *Pages 106–115 of: Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*. IEEE.

Liu, Ziwei, Luo, Ping, Wang, Xiaogang, & Tang, Xiaou. 2015. Deep Learning Face Attributes in the Wild. *In: Proceedings of International Conference on Computer Vision (ICCV)*.

Machanavajjhala, Ashwin, Kifer, Daniel, Gehrke, Johannes, & Venkitasubramaniam, Muthuramakrishnan. 2007. l-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1), 3.

Netzer, Yuval, Wang, Tao, Coates, Adam, Bissacco, Alessandro, Wu, Bo, & Ng, Andrew Y. 2011. Reading digits in natural images with unsupervised feature learning. *Page 5 of: NIPS workshop on deep learning and unsupervised feature learning*, vol. 2011.

Papernot, Nicolas, Abadi, Martin, Erlingsson, Úlfar, Goodfellow, Ian, & Talwar, Kunal. 2016. Semi-supervised knowledge transfer for deep learning from private training data. *arXiv preprint arXiv:1610.05755*.

Papernot, Nicolas, Song, Shuang, Mironov, Ilya, Raghunathan, Ananth, Talwar, Kunal, & Erlingsson, Úlfar. 2018. Scalable Private Learning with PATE. *arXiv preprint arXiv:1802.08908*.

Pérez-Cruz, Fernando. 2008. Kullback-Leibler divergence estimation of continuous distributions. *Pages 1666–1670 of: Information Theory, 2008. ISIT 2008. IEEE International Symposium on*. IEEE.

Schneider, Matthew J, & Abowd, John M. 2015. A new method for protecting interrelated time series with Bayesian prior distributions and synthetic data. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 178(4), 963–975.

Shokri, Reza, & Shmatikov, Vitaly. 2015. Privacy-preserving deep learning. *Pages 1310–1321 of: Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*. ACM.

Sweeney, Latanya. 2002. K-anonymity: A Model for Protecting Privacy. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, 10(5), 557–570.

Szegedy, Christian, Vanhoucke, Vincent, Ioffe, Sergey, Shlens, Jon, & Wojna, Zbigniew. 2016. Rethinking the inception architecture for computer vision. *Pages 2818–2826 of: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Xie, Liyang, Lin, Kaixiang, Wang, Shu, Wang, Fei, & Zhou, Jiayu. 2018. Differentially Private Generative Adversarial Network. *arXiv preprint arXiv:1802.06739*.

Zhang, Xinyang, Ji, Shouling, & Wang, Ting. 2018. Differentially Private Releasing via Deep Generative Model. *arXiv preprint arXiv:1801.01594*. 