Adversarial Learning Attack on Synthetic Data Evaluation

Kyrylo Rudavskyy
Ryerson University
Toronto, Canada

I. Introduction

Contemporary computing is facing twin, contradictory demands concerning data. On the one hand, with recent improvements in technology, machine learning is experiencing unprecedented growth. This growth entails a high demand for data. However, as people become more conscious of the externalities associated with data sharing, the demand for privacy also increases. The twin goals of abundant data and strong privacy are difficult to reconcile. The concept of synthetic data is an attempt to resolve this paradox.

There are two dominating approaches to synthesizing data, as demonstrated by the NIST synthetic data challange[6]. One approach is to build the joint distribution from the marginals [7]. This is a computationally efficient and precise method[7]. However, since this method already produces the distribution, further learning could be redundant.

The other approach is based on the Generative Adversarial Networks (GANs)[10]. This approach involves two neural networks. One is called the generator, and the other the critic. The generator alters the original by adding noise, while the critic tries to spot the difference. Eventually, the two networks arrive at equilibrium and a synthetic data set is produced.

This approach is computationally more difficult and less precise [5] then the one above. However, unlike the above, GANs do not produce the distribution [4]. Therefore, machine learning on synthetic data is not redundant.

Although synthetic data offers superior privacy protection, this is still not enough. For example, US census decided to upgrade synthetic data generation with a technology called differential privacy [2] [3]. This technology adds noise to the original data. However, this noise is mathematically manipulated so that it is possible to establish rigorous limits of disclosure risk [9] [12].

When synthetic data is produced, a common technique to evaluate its quality is to measure similarity to the original. This is accomplished using a metric called PMSE [17]. The metric's main principle is to use a machine learning classifier, such as a Decision Tree, to evaluate the similarity to the original. If the classifier cannot distinguish between the original and synthetic, then it can be claimed the synthesis was successful.

Although this approach is well established, it contains a significant weakness. A classifier can be attacked using adversarial learning[18]. This is especially true in a large-scale, industrial setting where synthesis is highly automated, and training samples can be easily manipulated. The aim of this paper is to evaluate the feasibility of such an attack. To the best of the author's knowledge, this was not done before.

II. Problem Statement and Data

The attack on the PMSE classifier will be carried out along the input vector. The original data will be poisoned to decrease PMSE. A smaller PMSE is an indicator of a better synthesis process. The topology of the attack can be seen in "Fig. 1".

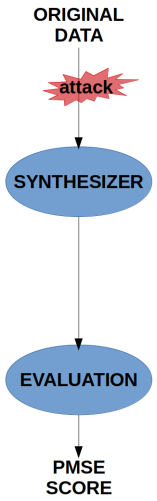

Fig. 1. Topology of the PMSE classifier attack.

The threats stemming from the attack are prominent if the number of altered data points is small and the poisoned data resembles closely the original. For example, let us assume the original data is collected via a distributed acquisition process, such as a mobile application. Then a malicious node could inject altered data points and compromise the entire production process.

Finally, this attack demonstrates a theoretic shortfall of the PMSE measure. A differentially private version of synthetic data contains an element of randomness[1]. This randomness could introduce noise similar to the one in the attack and change the PMSE score.

In principle, the data set choice is not important for the purpose of this work. However, the choice will be constrained to tabular data to avoid generalization issues. The specific

dataset chosen for the experiments is US Adult Census[1] dataset [8][11]. The set has the following characteristics:

- 48842 tuples and 15 features
- 5 numerical features and 7 categorical
- binary label: original or synthetic
- equal number of original and synthetic tuples

III. Methodology

The attack will be divided into the main stages, and each stage will be discussed in detail. The attack consists of the following steps: (1) data synthesis, (2) synthesis evaluation, (3) adversarial learning of the evaluation algorithm, (4) manipulated data synthesis, and (5) repeat evaluation with an improved score. Thus, the end goal is to improve the quality metric of the synthetic data.

A. Data Synthesis

The data is synthesized using a Wasserstein GAN with a Gradient Penalty (WGAN-GP) as described in [10]. WGAN-GP is a two-step improvement over vanilla GANs. First, Wasserstein distance is used in GAN architecture. Second, a gradient penalty is added to the objective function.

WGANs were introduced in [4]. The authors note that GANs do not learn the probability density. Instead, they generate samples from a distribution that approximates the real one. Moreover, GANs can train on high-dimensional data. The main advantage of WGANs is the use of Wasserstein distance to measure the distance between the two distributions. This measure has superior mathematical properties in the given context. Mainly, WGANs result in more robust and reliable training.

WGAN-GP architecture was introduced in [10]. The problem with WGANs above is that to solve the optimization problem, which approximates the real distribution, the authors in [4] clip the weights of the critic network. Per [10], clipping is done to ensure that the critic gradient norm is at most one and, thus, the gradient is differentiable. Differentiability is required to solve the loss function.

Instead of weight clipping, the WGAN-GP approach introduces a gradient penalty to the objective function to promote the necessary norm size. In practice, this approach has two benefits. It produces more stable gradients in very deep networks and approximates more complicated optimal functions. Because of these advantages, WGAN-GP architecture was chosen for this project. Finally, it is important to note that WGAN-GPs require the input data to be numeric and scaled.

B. Evaluation

Once the data is synthesized, its quality can be evaluated by measuring its semblance to the original. A common approach is to use a metric called PMSE, described in [17]. To calculate PMSE, the following steps are necessary. Original and synthetic data are assigned binary labels and mixed. Then, a decision tree classifier is trained. The classifier produces a vector of prediction probabilities $\vec{p}$. Each entry in the vector corresponds to a tuple of data. Finally, the following equation is applied

$$PMSE = \frac{1}{n}\sum_{i=0}^{n}(p_i - c)^2 \tag{1}$$

where c is the probability of a tuple being synthetic. i.e. number of synthetic tuples divided by n synthetic and original tuples.

The critical observation here is that PMSE decreases proportionally to the tree's inability to correctly classify tuples. This fact is key to the proposed attack.

C. Adversarial Learning

Adversarial learning is a machine learning technique with a purpose that is opposite in spirit to regular learning. The goal of adversarial learning is to produce samples that force a machine-learning algorithm to make erroneous classifications. The samples are created by introducing imperceptible variations into real samples.

This work relies on a technique called Decision Tree Attack proposed in [15]. This technique exploits the tree structure to reverse engineer sample modifications that will result in misclassification. It takes as input a tree classifier, sample of data and predicted labels. Once processed, it produces an adversarial sample. The similarity between adversarial and input samples can be scaled using parameter tuning.

D. Classifier Attack

With the above tools at hand, it is possible to describe the entire process("Fig. 2"). Data is synthesized using a WGAN-GP and evaluated with a PMSE. During the evaluation, a subset is used for training and the rest for testing. This work chose a 50/50 split. The training samples will be discarded and must be possessed by the adversary. The testing sample will be used to create an adversarial data set.

The testing set, the classifier, and the testing set's predictions are passed to the Decision Tree Attack (DTA) algorithm. The predictions are rounded into a deterministic, binary format. DTA slightly modifies the testing set to produce classification errors. Since the label is binary, DTA has the effect of toggling predictions. This creates the necessary confusion to decrease the PMSE score. A lower PMSE score entails a higher synthetic data quality.

However, before submitting the adversarial testing sample for the second PMSE evaluation, a new synthetic data set must be produced from the adversarial set and evaluated using a new PMSE instance. Therefore, for the attack to work, WGAN-GP must reproduce the sample modifications created by DTA. Moreover, adversarial learning must transfer from the previous PMSE instance.

If the amount of noise introduced by DTA is subtle, it is reasonable two assumes that the learning will transfer between two trees trained on similar data. However, it is less evident that the GAN-based synthesizer will reproduce the noise.

For this reason, WGAN-GP was chosen. Its ability to build deep networks and model complex functions may reproduce

[1]https://archive.ics.uci.edu/ml/datasets/adult

the DTA noise in the synthetic version. Herein lies an interesting paradox. If the above is true, then increasing the sophistication of synthetic data production makes it more susceptible to adversarial attacks.

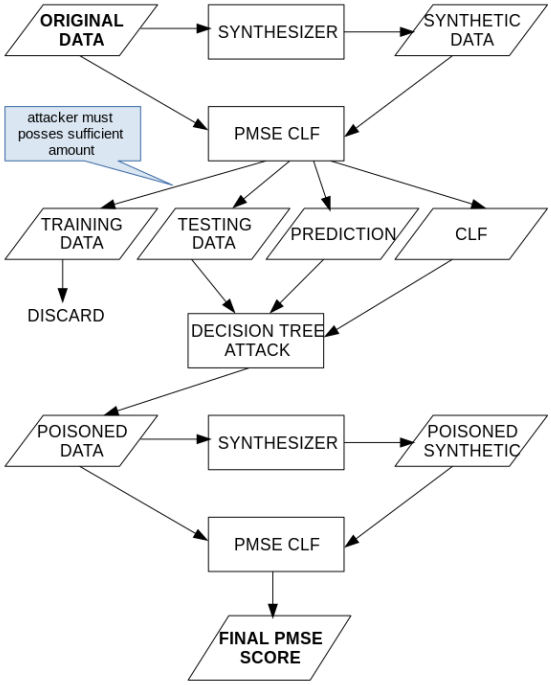

Fig. 2. Anatomy of data synthesis and evaluation attack.

IV. Results

Experimental results confirmed the above assumptions and demonstrated success. The attack was performed 20 times with the following outcome:

- PMSE score decreased by 20%
- 5 out of 60 features were modified
- 3% of data points were altered

The number of data points modified was small enough for transfer learning to occur between two trees. However, it was large enough for WGAN-GP to reproduce the DTA noise. The boxenplots[2] of the clean testing sample and the adversarial one does not demonstrate inconsistent alterations ("Fig. 3").

The values stay within the bounds of the original. Outliers are similar to the ones in the original. A noticeable difference is that the age structure changed. This can be seen in the bottom row of "Fig. 3". The new human population is much younger than in the original. A domain expert could notice such a change.

Another noticeable difference is that some categorical features became non-deterministic. Such a change can be expected from a GAN-based synthesis as well. Thus, non-deterministic categorical variables don't need to pose a threat to attack validity.

[2]https://seaborn.pydata.org/generated/seaborn.boxenplot.html

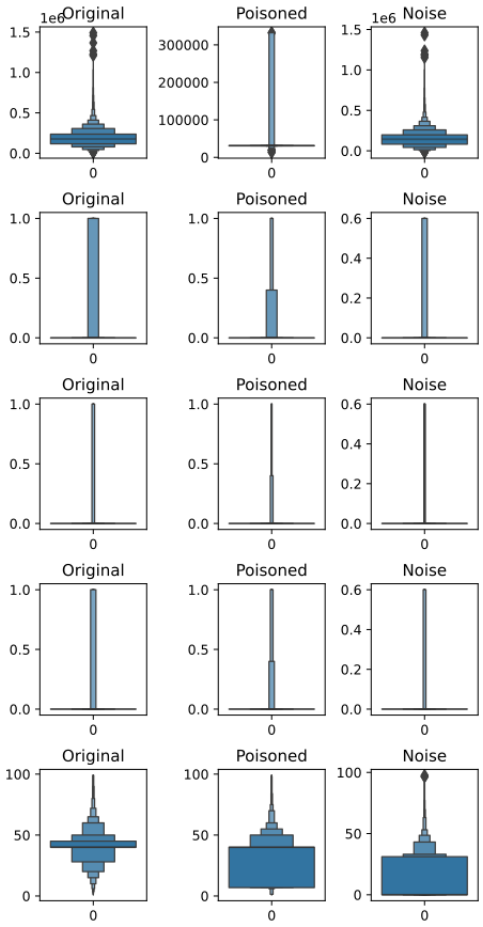

Fig. 3. Features altered by the attack.

V. Discussion

Originally this attack was designed for a differentially private version of GAN-based synthesis. However, a general-purpose implementation of a DP version was not available. The only DP WGAN available[3] was tailor-made for a large dataset and required hours to process. Thus practical experimentation was impossible without significant computing resources. Also, in the absence of a more advanced GAN architecture, it is unclear if the synthesizer would transfer DTA noise.

In case the simpler versions of GANs do not reproduce the DTA noise, the attack can be modified. A space of adversarial samples can be established from the DTA. Then, a GAN-based synthesizer could be attacked directly to produce output within that space.

To better judge the attack's validity, it is necessary to establish the amount of data an adversary needs to construct a decision tree. Although this experiment used a 50/50 split, the actual number should be much smaller. This is likely a function of dataset size. However, a theoretical boundary of the raining sample size should be possible to establish.

[3]https://www.nist.gov/ctl/pscr/team-uclanesl

The attack was performed on one data set of tabular type. To further investigate the limits of this attack, it is necessary to test it on a wide range of data sets. Also, other GAN-based synthesis methods should be tried. Finally, non-tabular data should be tested as well.

Having established the limits of the attack, it is necessary to follow-up with countermeasures. A simple approach could be to validate poisoned data with a domain expert. However, this hinges on the propensity of DTA to change the fundamental characteristics of data. Also, not all data is subject to domain expertise.

Although decision trees are a popular choice, other classifiers should be used for PMSE and attacked. This will completely exhaust the question of attacks against synthetic data evaluation.

VI. Implementation

This work was implemented in Python 3 using Numpy and Pandas for data handling. Other packages used will be listed below. All are open-source and available for Python.

The data was fetched using the Penn Machine Learning Benchmarks package[14]. This is a large python library of datasets. The sets are divided into classification and regression categories. Sets can also be selected by feature type - continuous, ordinal, categorical - or by the number of target classes.

The WGAN-GP implementation was provided by the YData[4]. It is a private company that provides data processing solutions. Its synthesizers are available on GitHub.

The implementation of the DTA was taken from the Adversarial Robustness Toolbox v1.0.0 package[13]. It is a large research project created to provide a library of well known adversarial learning attacks and countermeasures.

Finally, the decision tree was implemented using scikit-learn library[16]. The same package was used for processing categorical data and scaling numerical.

References

[1] Martín Abadi et al. *Deep Learning with Differential Privacy*. Oct. 24, 2016. arXiv: 1607.00133 [cs, stat]. URL: http://arxiv.org/abs/1607.00133 (visited on 12/14/2020).

[2] John M Abowd. "Research Data Centers, Reproducible Science, and Confidentiality Protection: The Role of the 21st Century Statistical Agency". In: (), p. 60.

[3] John M. Abowd. "The U.S. Census Bureau Adopts Differential Privacy". In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. KDD '18: The 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. London United Kingdom: ACM, July 19, 2018, pp. 2867–2867. ISBN: 978-1-4503-5552-0. DOI: 10.1145/3219819.3226070. URL: https://dl.acm.org/doi/10.1145/3219819.3226070 (visited on 05/22/2020).

[4] Martin Arjovsky, Soumith Chintala, and Léon Bottou. *Wasserstein GAN*. Dec. 6, 2017. arXiv: 1701.07875 [cs, stat]. URL: http://arxiv.org/abs/1701.07875 (visited on 12/13/2020).

[5] Claire McKay Bowen and Joshua Snoke. *Comparative Study of Differentially Private Synthetic Data Algorithms and Evaluation Standards*. Nov. 28, 2019. arXiv: 1911.12704 [cs, stat]. URL: http://arxiv.org/abs/1911.12704 (visited on 06/05/2020).

[6] brianna.vendetti@nist.gov. *2018 Differential Privacy Synthetic Data Challenge*. NIST. URL: https://www.nist.gov/ctl/pscr/open-innovation-prize-challenges/past-prize-challenges/2018-differential-privacy-synthetic (visited on 05/30/2020).

[7] Rui Chen et al. "Differentially Private High-Dimensional Data Publication via Sampling-Based Inference". In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '15. Sydney, NSW, Australia: Association for Computing Machinery, Aug. 10, 2015, pp. 129–138. ISBN: 978-1-4503-3664-2. DOI: 10.1145/2783258.2783379. URL: https://doi.org/10.1145/2783258.2783379 (visited on 06/03/2020).

[8] Dheeru Dua and Casey Graff. "UCI Machine Learning Repository". In: (2017). URL: http://archive.ics.uci.edu/ml.

[9] Cynthia Dwork. *A firm foundation for private data analysis*. English. 2011.

[10] Ishaan Gulrajani et al. *Improved Training of Wasserstein GANs*. Dec. 25, 2017. arXiv: 1704.00028 [cs, stat]. URL: http://arxiv.org/abs/1704.00028 (visited on 12/13/2020).

[11] Ron Kohavi. "Scaling Up the Accuracy of Naive-Bayes Classifiers: A Decision-Tree Hybrid". In: (1996), p. 6.

[12] Jaewoo Lee and Chris Clifton. "How Much Is Enough? Choosing Epsilon for Differential Privacy". In: *Information Security*. Ed. by Xuejia Lai, Jianying Zhou, and Hui Li. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2011, pp. 325–340. ISBN: 978-3-642-24861-0. DOI: 10.1007/978-3-642-24861-0_22.

[13] Maria-Irina Nicolae et al. *Adversarial Robustness Toolbox v1.0.0*. Nov. 15, 2019. arXiv: 1807.01069 [cs, stat]. URL: http://arxiv.org/abs/1807.01069 (visited on 12/07/2020).

[14] Randal S. Olson et al. "PMLB: A Large Benchmark Suite for Machine Learning Evaluation and Comparison". In: *BioData Mining* 10.1 (Dec. 11, 2017), p. 36. ISSN: 1756-0381. DOI: 10.1186/s13040-017-0154-4. URL: https://doi.org/10.1186/s13040-017-0154-4 (visited on 12/15/2020).

[15] Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. *Transferability in Machine Learning: From Phenomena to Black-Box Attacks Using Adversarial Samples*. May 23, 2016. arXiv: 1605.07277 [cs]. URL: http://arxiv.org/abs/1605.07277 (visited on 10/01/2020).

[4] https://github.com/ydataai/ydata-synthetic

[16] F. Pedregosa et al. “Scikit-learn: Machine Learning in Python”. In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[17] Joshua Snoke et al. *General and Specific Utility Measures for Synthetic Data*. June 18, 2017. arXiv: 1604.06651 [stat]. URL: http://arxiv.org/abs/1604.06651 (visited on 06/16/2020).

[18] Jingwen Zhao, Yunfang Chen, and Wei Zhang. “Differential Privacy Preservation in Deep Learning: Challenges, Opportunities and Solutions”. In: *IEEE Access* 7 (2019), pp. 48901–48911.