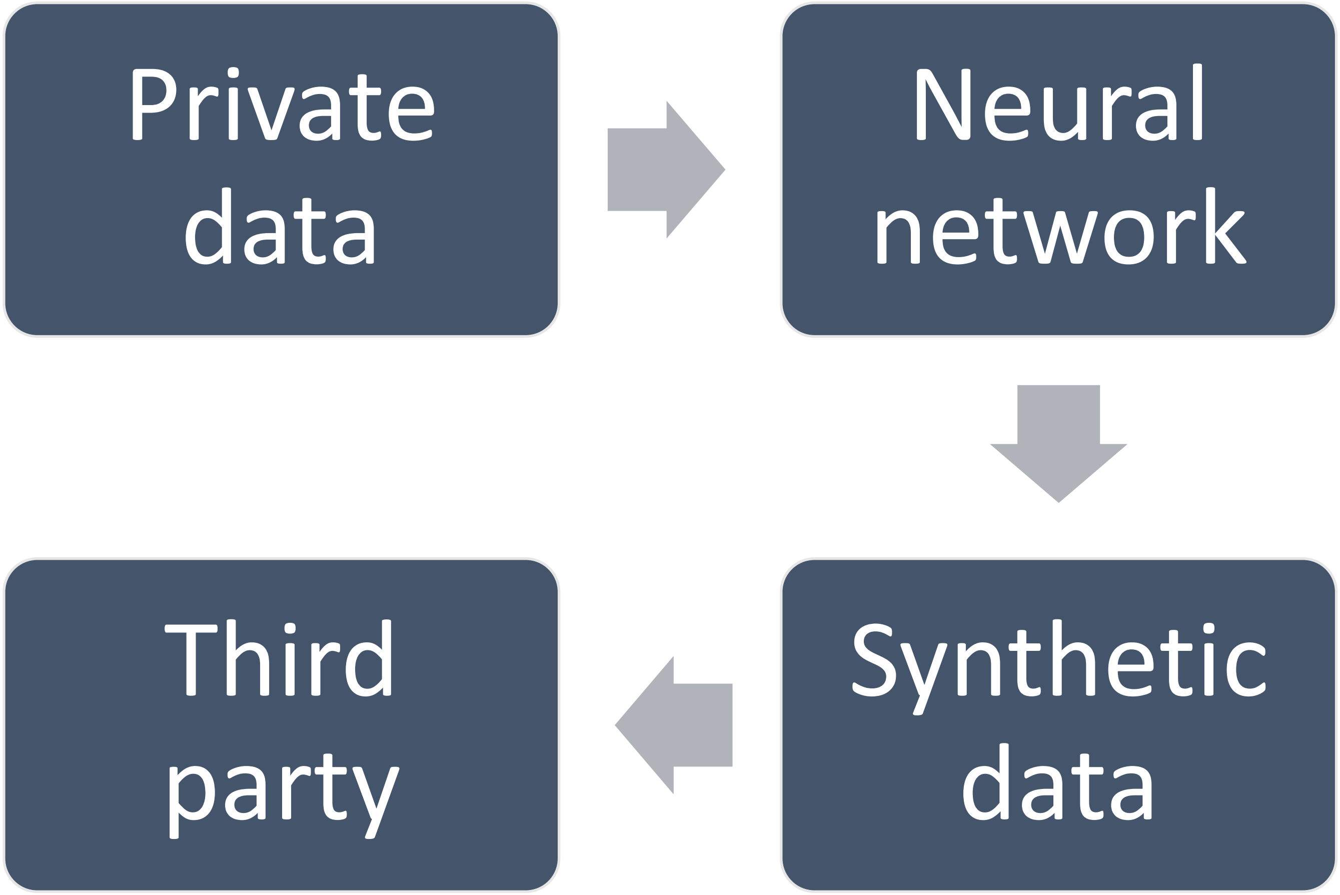
Robustness of GAN-based image generators against adversarial attacks

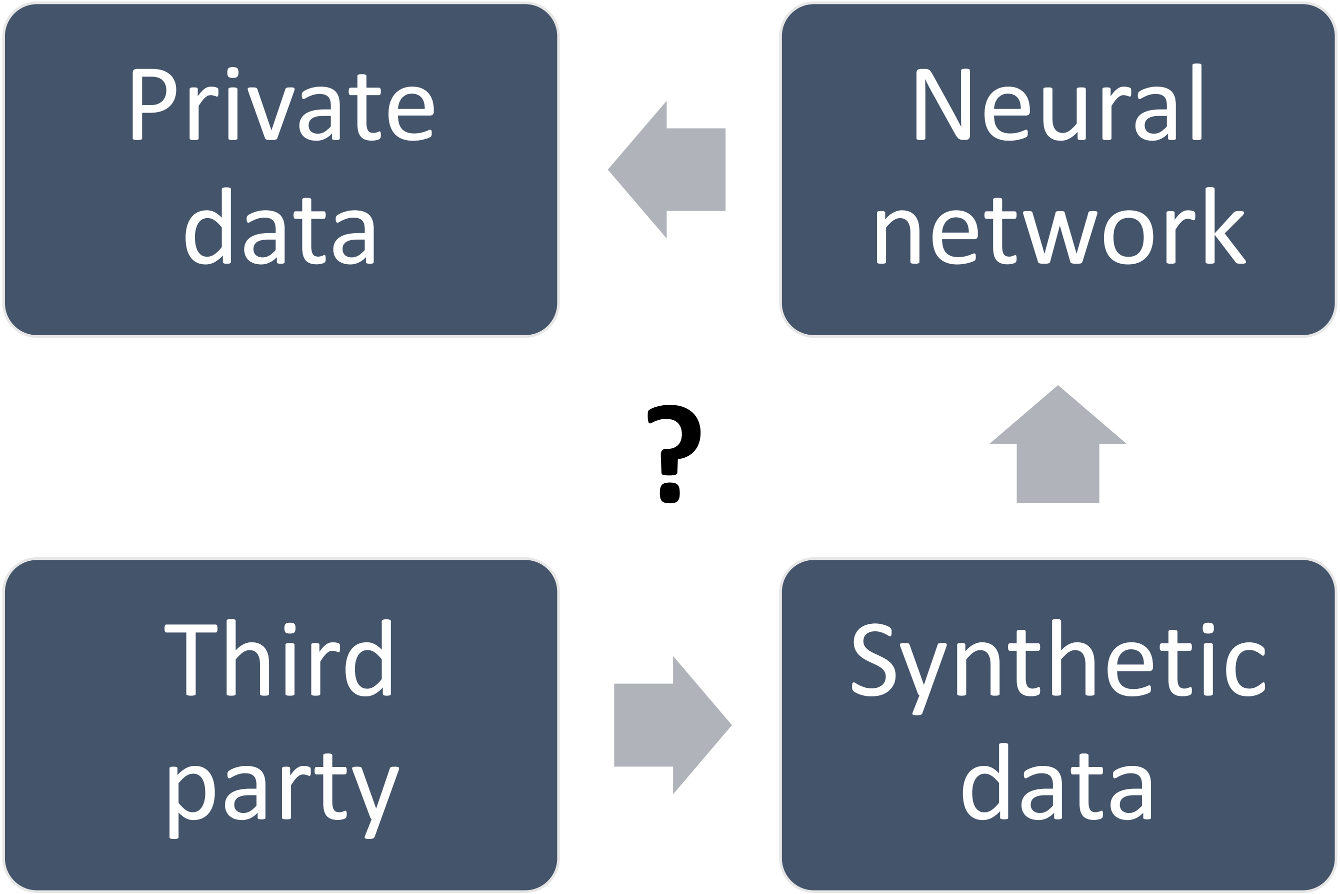
Artur Sidorenko[1]; Natalya Denisenko[2]; Alexey Mironov[2]; Denis Derkach, PhD[2]; Nikita Kaseev[2]; Andrey Ustushanin[2]

[1]Moscow State University, [2]Higher School of Economics

Private datasets cannot be outsourced, but one can generate synthetic data via a GAN


Private data
Neural network
Synthetic data
Third party

An adversary can subvert the GAN to get information about the private dataset



Private data

Neural network

?

Third party

Synthetic data

GANs are highly effective, but they should not disclose private information

Generative Adversarial Network (GAN) has proven to be a highly efficient technology. It shows remarkable results in a wide range of applications.

However, sufficient and representative datasets are prerequisite for successful training of a GAN. These datasets are often crowdsourced and outsourced and may contain private information. These models must not expose private information because disclosure of such data is treated as a breach of law.

Cheap but good solutions can be helpful

In this research study, we investigate attacks against generative models proposed in [1]: given a record, the adversary determines if it was present in the training dataset. These attacks are aimed at detecting any disclosure of personal data.

We have discovered that even a simplistic GAN can be potent to secure private information if one follows recommendations given in [1].

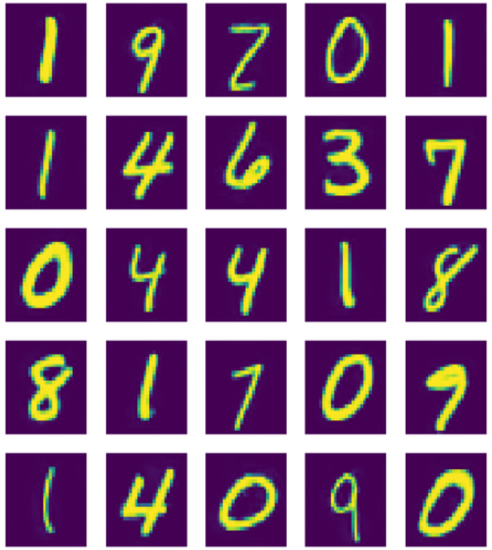
1. J. Hayes , L. Melis , G. Danezis, E. De Cristofaro. *LOGAN: Membership Inference Attacks Against Generative Models.* Proceedings on Privacy Enhancing Technologies, 2019(1), pp.133-152
5. Pretrained DCGAN for MNIST: https://github.com/csinva/gan-vae-pretrained-pytorch/tree/master/mnist_dcgan

Research structure: training two networks and imitating different types of attacks

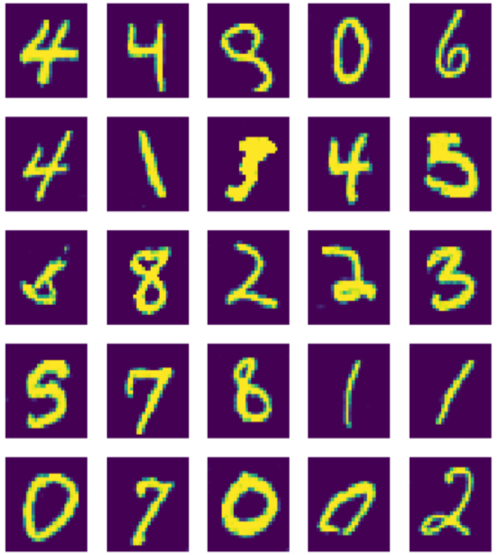
- **Training** two neural networks: DCGAN [5] and Vanilla GAN (a simple sequence of convolutional layers).
- **Imitating** typical attacks: white-box, black-box with and without leakage.
  - **White-box attack**: given discriminator reveal if a given image is a part of the training set.
  - **Black-box attack with leakage**: training a home-made classifier using leakage and artificially generated images.
  - **Black-box attack without leakage**: training a home-made GAN.

1. J. Hayes , L. Melis , G. Danezis, E. De Cristofaro. *LOGAN: Membership Inference Attacks Against Generative Models.* Proceedings on Privacy Enhancing Technologies, 2019(1), pp.133-152
5. Pretrained DCGAN for MNIST: https://github.com/csinva/gan-vae-pretrained-pytorch/tree/master/mnist_dcgan

We trained a simplistic GAN and employed a pre-trained DCGAN [5]



Vanilla GAN



DCGAN

The structure of our vanilla GAN

Generarator

- Dense (ELU)
- Dropout
- Conv2d (kernel_size=3, ELU)
- Dropout
- ConvTranspose2d
- Dropout
- Conv2d и Dropout
- Conv2d и Dropout

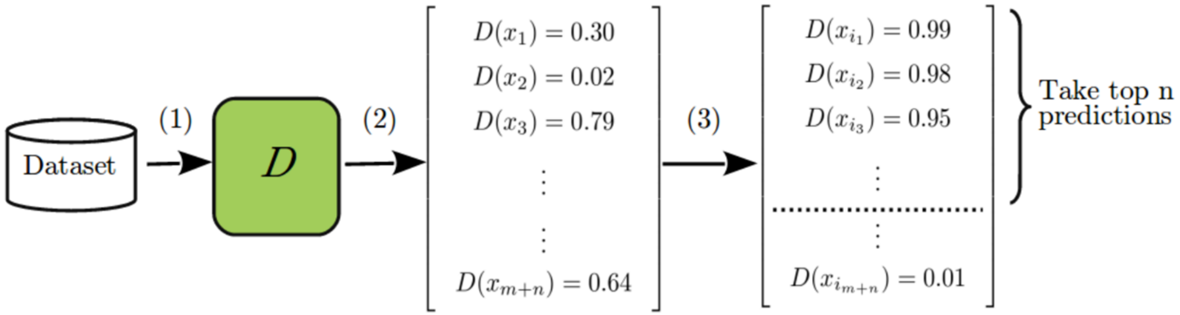
Discriminator

- Conv2d (ELU, kernel_size=3)
- Dropout
- MaxPool2d(2)
- Conv2d(ELU,kernel_size=3) и Dropout
- Conv2d(ELU,kernel_size=3) и Dropout
- Conv2d(ELU,kernel_size=3) и Dropout
- Linear() with sigmoid

1. J. Hayes , L. Melis , G. Danezis, E. De Cristofaro. *LOGAN: Membership Inference Attacks Against Generative Models.* Proceedings on Privacy Enhancing Technologies, 2019(1), pp.133-152

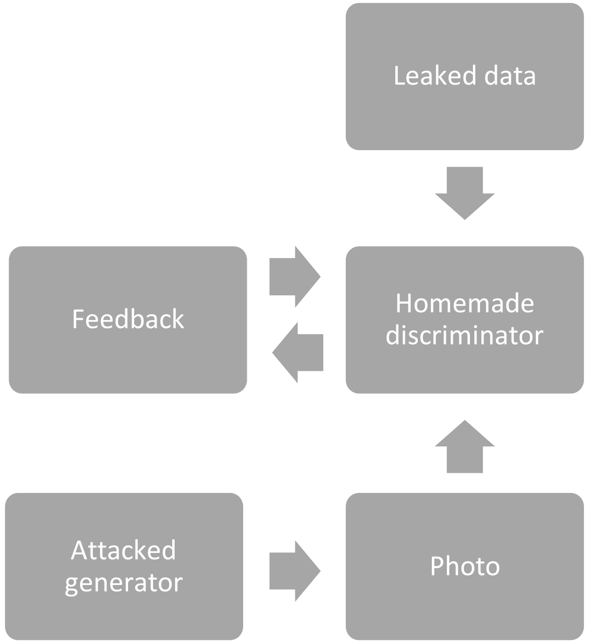
5. Pretrained DCGAN for MNIST: https://github.com/csinva/gan-vae-pretrained-pytorch/tree/master/mnist_dcgan

Each type of attack requires certain leakage of data



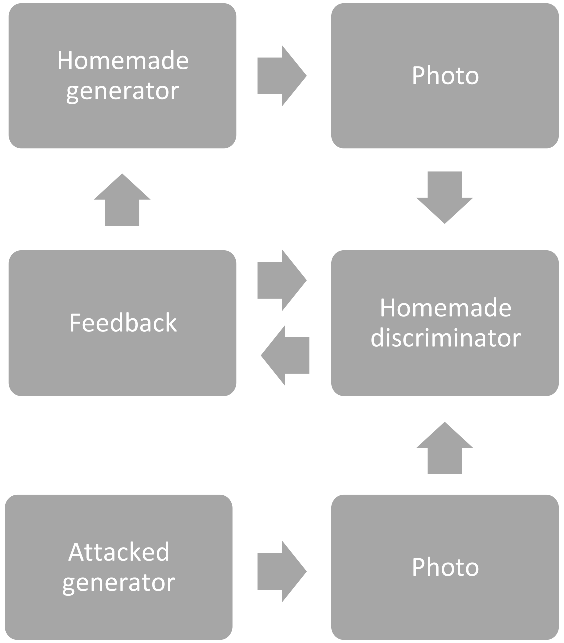
White-box: an adversary stole the discriminator

(the image is taken from [1])



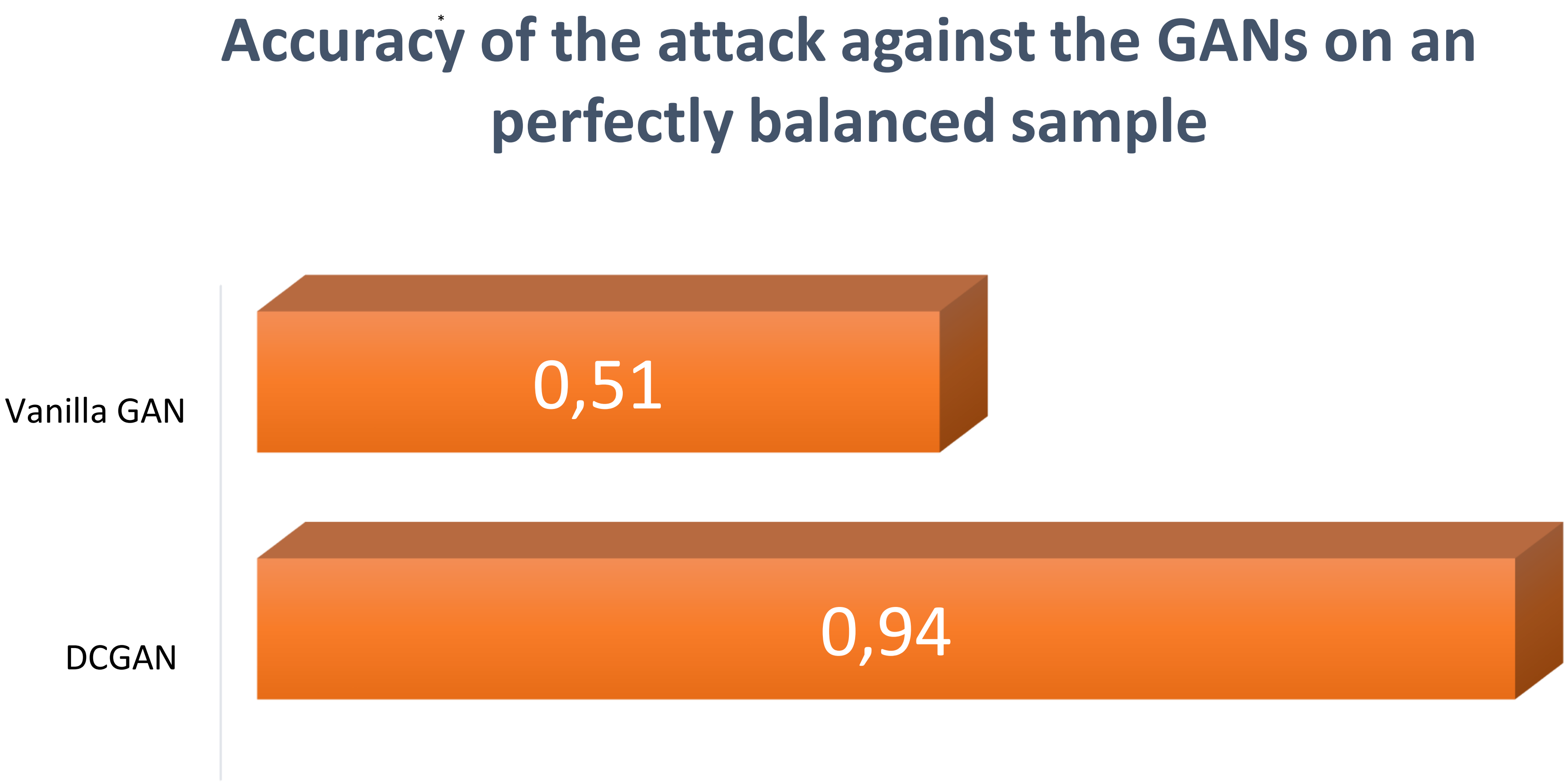
Black-box attack + leakage of private data

Black-box attack without leakage: the structure of GAN is required for a successful attack

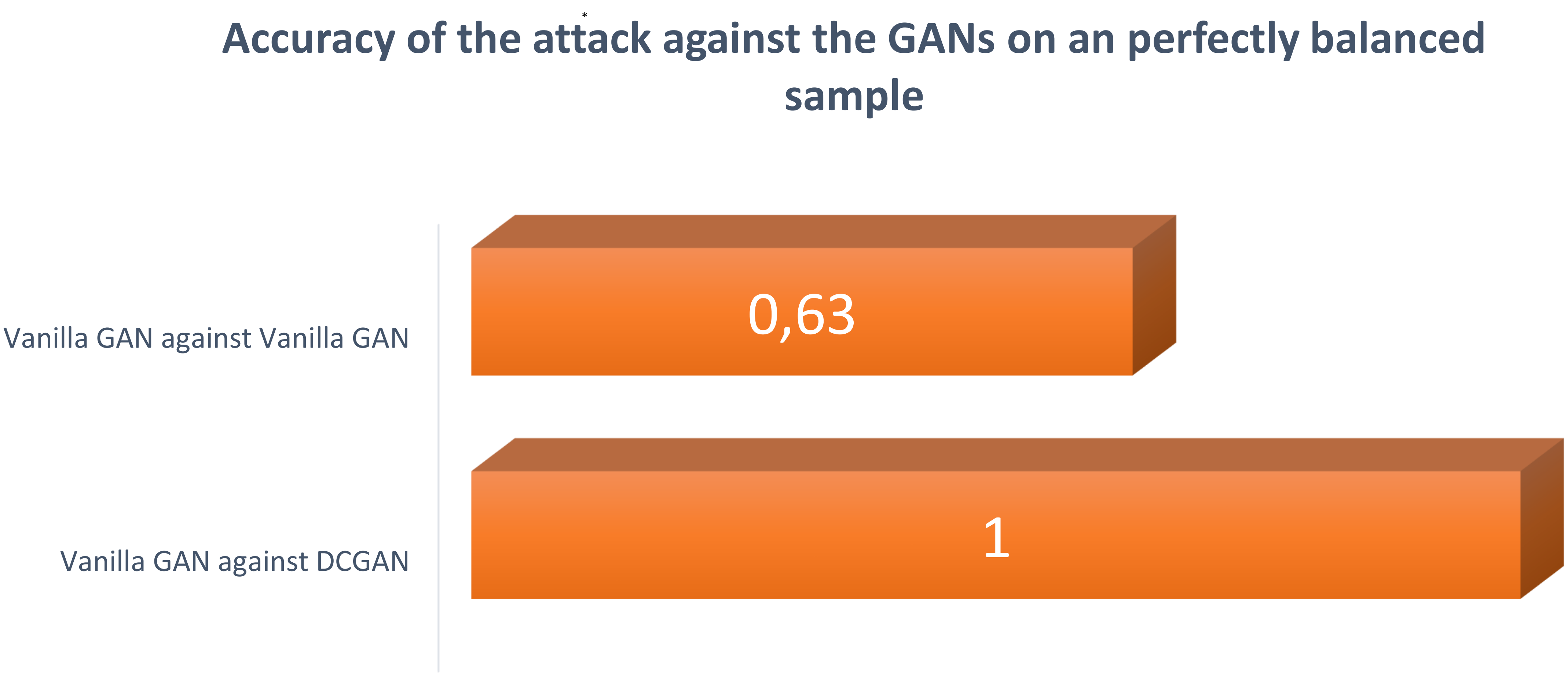


1. J. Hayes , L. Melis , G. Danezis, E. De Cristofaro. *LOGAN: Membership Inference Attacks Against Generative Models.* Proceedings on Privacy Enhancing Technologies, 2019(1), pp.133-152

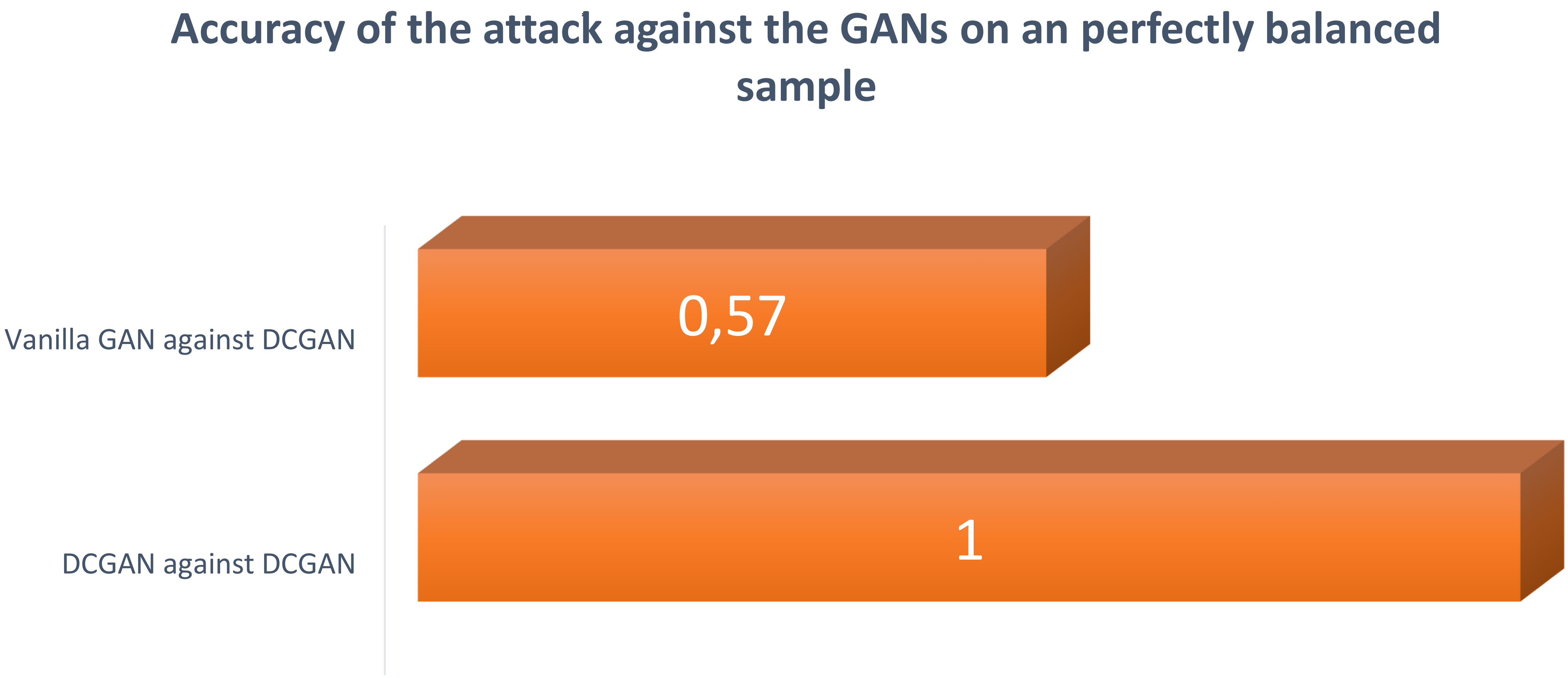
Even simplistic GANs are robust against white-box attacks



Even simplistic GANs are robust against black-box attacks (20% leakage)



Black-box attacks without leakages and knowledge about attacked GAN are ineffective



Results

Both Vanilla GAN and DCGAN replicate handwritten digits at the same quality. Vanilla GAN, however, demonstrates extreme robustness. DCGAN can be easily subverted once its structure is uncovered. Despite straightforwardness of the white-box attack, black-box attacks can also show high efficiency if particular information (the structure of the GAN or a fraction of the dataset itself) has flown to the adversary.

Simple structures show high robustness against them if one follows recommendations given in [1]: avoidance of overfitting, placing dropout layers, implementing proposed attacks before committing the network.

References

1. J. Hayes , L. Melis , G. Danezis, E. De Cristofaro. *LOGAN: Membership Inference Attacks Against Generative Models.* Proceedings on Privacy Enhancing Technologies, 2019(1), pp.133-152
2. J. Jordon, J. Yoon, M. van der Schaar. *PATE-GAN: Generating synthetic data with differential privacy guarantee., https://openreview.net/pdf?id=S1zk9iRqF7 , 2019*
3. M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang. *Deep learning with differential privacy*. In CCS, 2016.
4. M. Arjovsky, S. Chintala, and L. Bottou. *Wasserstein GAN*. https://arxiv.org/pdf/1701.07875.pdf%20http://arxiv.org/abs/1701.07875.pdf , 2017.
5. Pretrained DCGAN for MNIST: https://github.com/csinva/gan-vae-pretrained-pytorch/tree/master/mnist_dcgan

Thanks for your attention!

Artur Sidorenko

Moscow State University, Faculty of Mechanics of Mathematics

a.sidorenko1998@yandex.ru

Differential privacy

- Two datasets D and D’ are said to be neighbouring if there is such x in D that $D \setminus \{x\} = D'$.
- Let $(\,\Omega, \mathcal{F}, P)$ be a probability space, let (X, S) be a measurable space representing datasets, and let (Y, T) be a measurable space of outcomes.
- Let M be a *randomised algorithm*, i.e. a measurable mapping

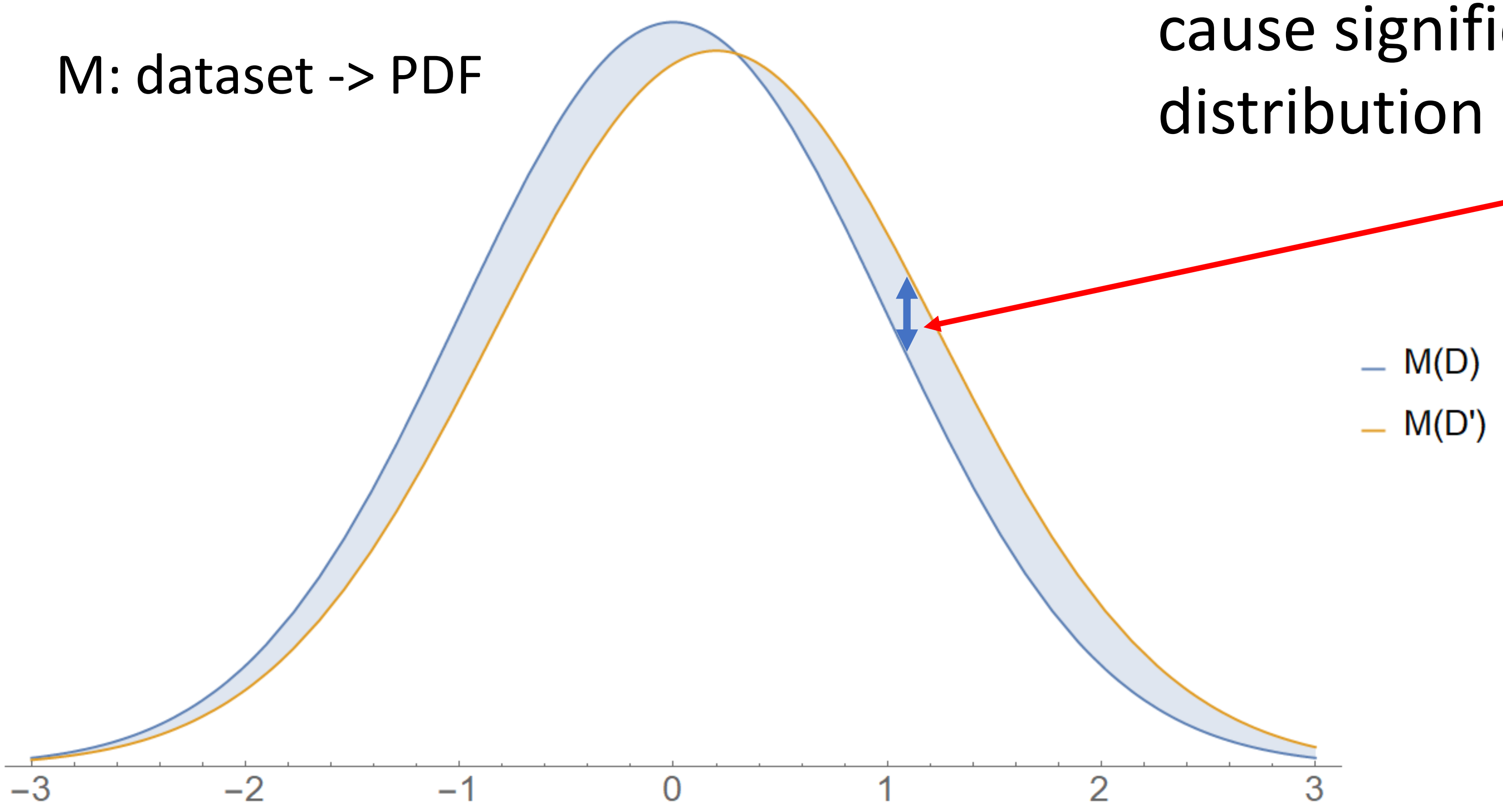
$$M: X \times \Omega \to Y$$

The algorithm is said to be (ε, δ)-private, if for all neighbouring datasets D and D’ and for all sets $Y \in T$: $P(M(D) \in Y) < e^{\varepsilon} P(M(D') \in Y) + \delta$

Differential privacy

M: dataset -> PDF

Slight changes of the dataset do not cause significant difference in the distribution of the output


M(D)
M(D')
−3
−2
−1
0
1
2
3