On the Generation and Evaluation of Synthetic Data with GANs

Bauke Brenninkmeijer
2019

Outline

- Introduction
- Preliminaries
- Tabular GANs
- Contributions
- Results

Problem

1. Synthesizing tabular data with GANs is difficult.
   - Different data types (continues, discrete)
   - Multi-modal data
   - Might contain long distance relationships
2. Evaluation of synthetic data is hard and performed inconsistent. What is ‘Good’ synthetic data? Different approaches in literature.

Motivation

Realistic synthesized data is very valuable for any domain where flow of data is restricted due to privacy, like in governments, healthcare and finance.

Goal

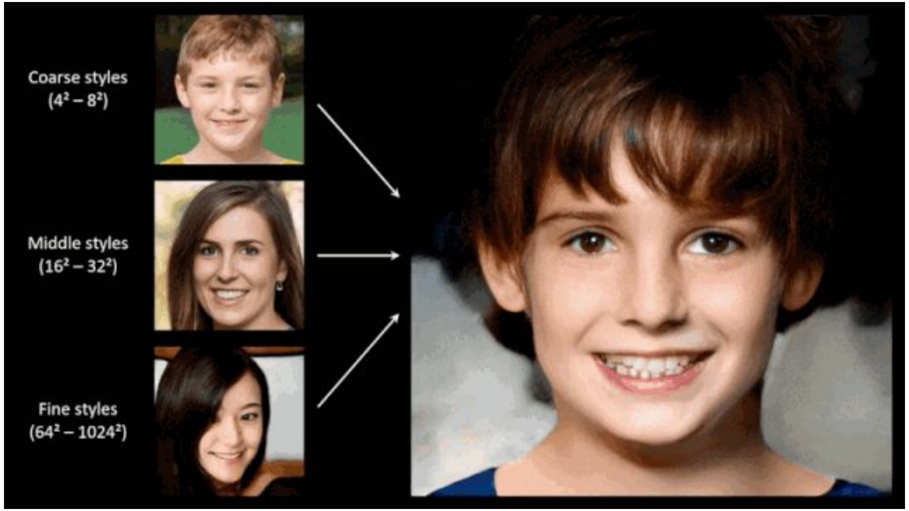
The Goal is twofold:

1. Improve the State-of-the-art for tabular data generation with GANs
2. Create a improved evaluation method that covers all aspects of data

Why Generative Adversarial Networks (GANs)?

Simple. They are the best for generating high dimensional data.

Clear in domains of images, audio and video



Preliminaries

Neural Networks

- Several layers that perform calculations
- Can be thought of as a function with parameters, which are trained to map domain X to Y.
- Results in one final layer, whose values can be trained to represent many things
- If the network has many layers, this is often referred to as deep learning

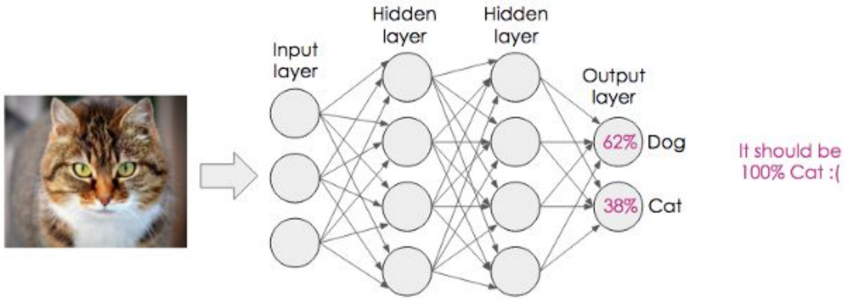
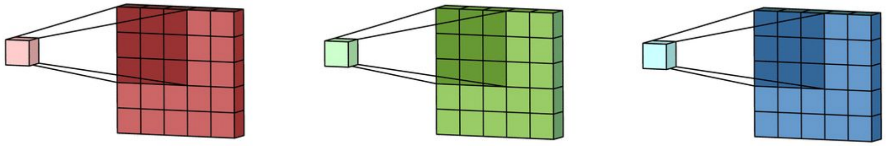
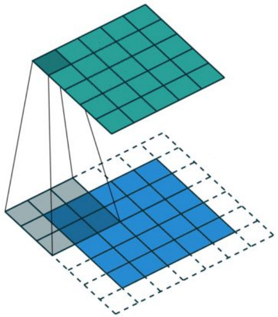


Image from: https://medium.com/@RosieCampbell/demystifying-deep-neural-nets-efb726eae941

Convolutional Neural Networks (CNNs)

- Finds specific structures
  High activation with cat face,
  low with dog face
- Translation Invariant
  Cat face in left corner and right
  corner cause similar reaction
- Many filters that capture
  patterns

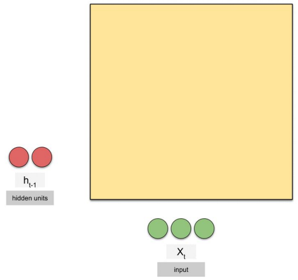


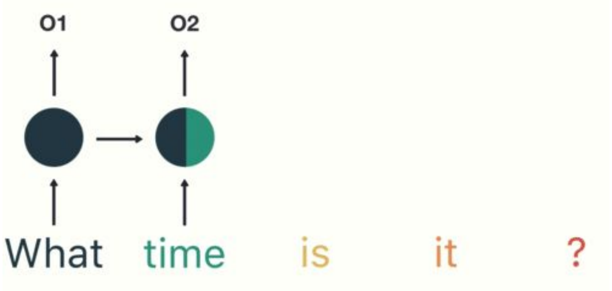


Recurrent Neural Networks (RNNs)

- Way to deal with sequential data
  Timeseries or text

- Often done with either GRU of LSTM
  Both have "internal memory"

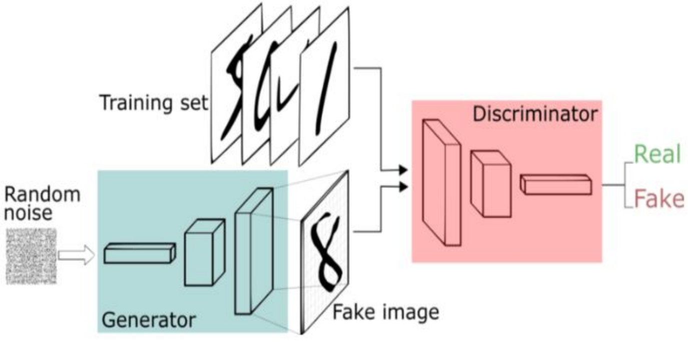




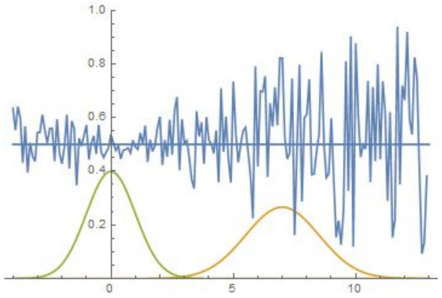
GANs in 5 minutes

Generative Adversarial Nets[1]

- Generator tries to imitate real data
- Discriminator tries to distinguish fake from real
- Minimax game between generator and Discriminator



Generative Adversarial Network



1 Goodfellow, Ian, et al. "Generative adversarial nets." Advances in neural information processing systems. 2014.

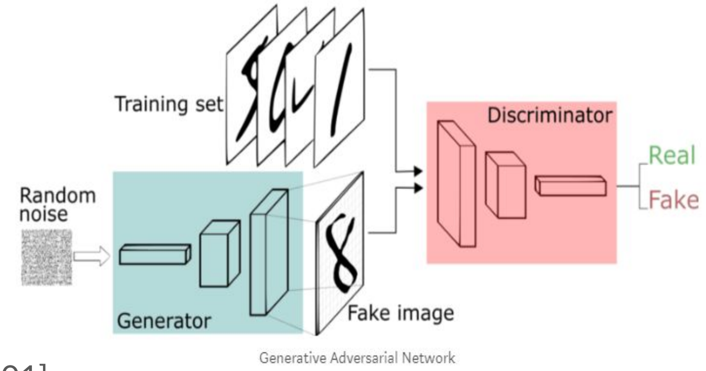
GANs in 5 minutes

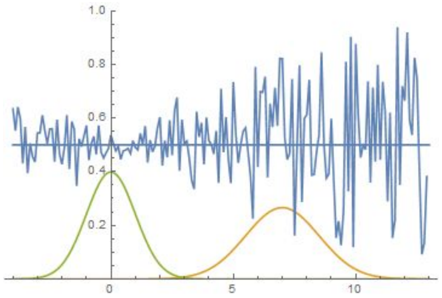
Wasserstein GAN[1]

- Uses 1-Wasserstein Distance
- Bring logit distributions of last layer closer together
- To enforce 1-Lipschitz constraint, clips critic weights to [-0.01, 0.01]
- Less mode collapse

WGAN-GP[2]

- Introduces gradient penalty instead of weight clipping
- Converges better, faster training





1 Arjovsky, Martin, Soumith Chintala, and Léon Bottou. "Wasserstein gan." arXiv preprint arXiv:1701.07875 (2017).
2. Density gif from https://monnoroch.github.io/posts/2018/04/03/generative-adversarial-networks.html

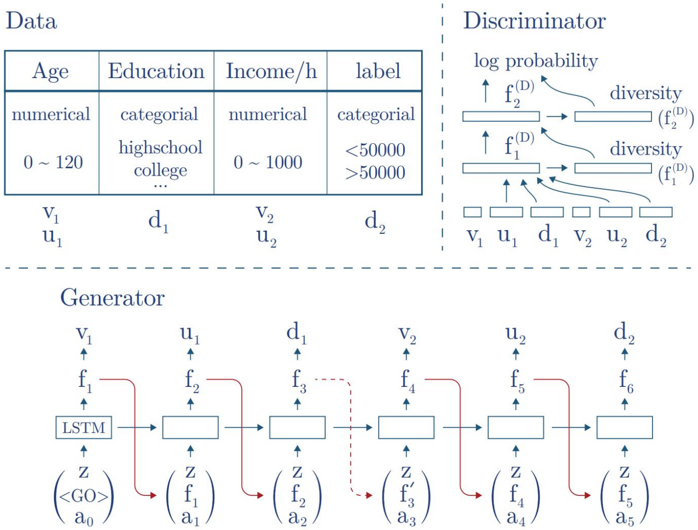
Related Work

Related work 1: TGAN[1]

- Encodes continuous values with Gaussian Mixture Models.
- Recurrent Neural Network
- Has an attention mechanism

Problems:

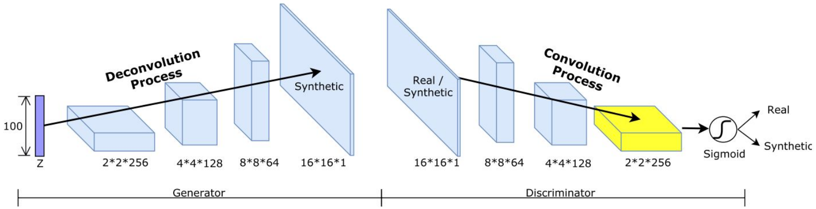
- Uses classic GAN Architecture



1 Xu, Lei, and Kalyan Veeramachaneni. "Synthesizing tabular data using generative adversarial networks." arXiv preprint arXiv:1811.11264 (2018).

Related work 2: TableGAN[1]

- Uses DCGAN, SotA method for images
- Handles ‘all’ datatypes
- Has classifier, identical to the discriminator, for semantic coherence as third GAN part



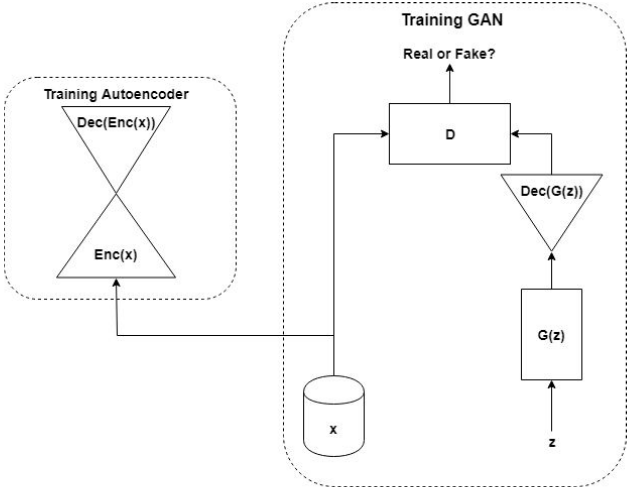
Problems

- Everything is represented as a float [0, 1]. Not ideal for discrete values.
- Also uses classic GAN architecture

1 Park, Noseong, et al. "Data synthesis based on generative adversarial networks." Proceedings of the VLDB Endowment 11.10 (2018): 1071-1083.

Related work 3: MedGAN[1]

- Autoencoders translates categorical features to continuous ones
- Circumvent categorical features in the generator



Problems

- Sampling autoencoder latent space is arbitrary for locations unseen during training
- Original implementation does not work with multiple data types

1 Park, N., Mohammadi, M., Gorde, K., Jajodia, S., Park, H., & Kim, Y. (2018). Data synthesis based on generative adversarial networks. Proceedings of the VLDB Endowment, 11(10), 1071-1083.

Related work: Evaluations

Baselines are all over the place

- Compared with data perturbation/anonymization tools
  sdcMicro, ARX
- Statistical sampling techniques
  Gaussian Copula, Bayesian Networks
- Neural approaches
  Boltzmann Machines, Variational autoencoders
- Humans experts

Evaluation approaches differ:

- Statistical evaluations
- Privacy evaluations

Tabular GANs

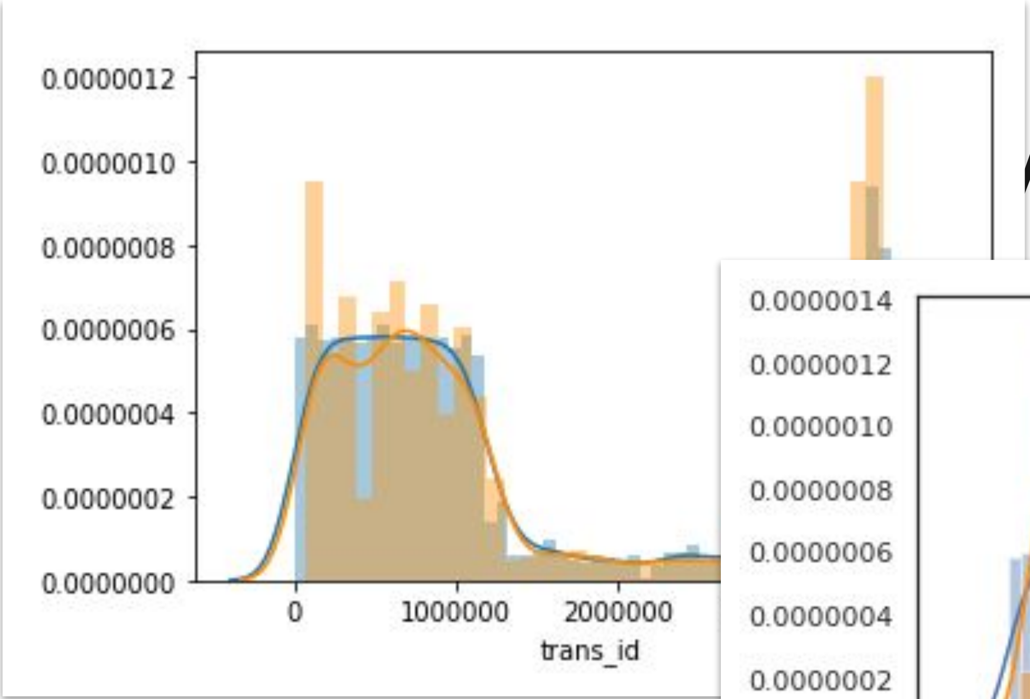
Lets get to it!

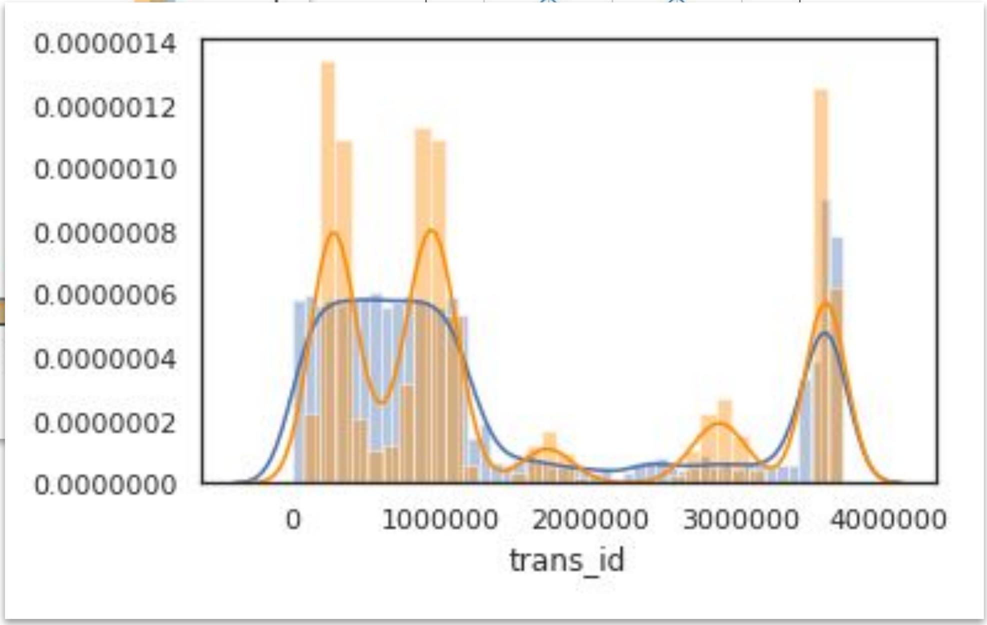
Hold on just a minute...

Types of data

- Continues data
- Categorical data
- Ordinal data

Values


0.0000012
0.0000010
0.0000008
0.0000006
0.0000004
0.0000002
0.0000000
0
1000000
2000000
trans_id


0.0000014
0.0000012
0.0000010
0.0000008
0.0000006
0.0000004
0.0000002
0.0000000
0
1000000
2000000
3000000
4000000
trans_id

Data Encoding: Categorical values

- Still often done using one-hot encoding
- In classifiers, success using embeddings
- Inverse transformation becomes very expensive. Requires distance measure which are often ambiguous in high dimensions. Options: Euclidean distance, cosine similarity, etc.

One-hot encoding

		cat	mat	on	sat	the
the	=>	0	0	0	0	1
cat	=>	1	0	0	0	0
sat	=>	0	0	0	1	0
…				…		

A 4-dimensional embedding

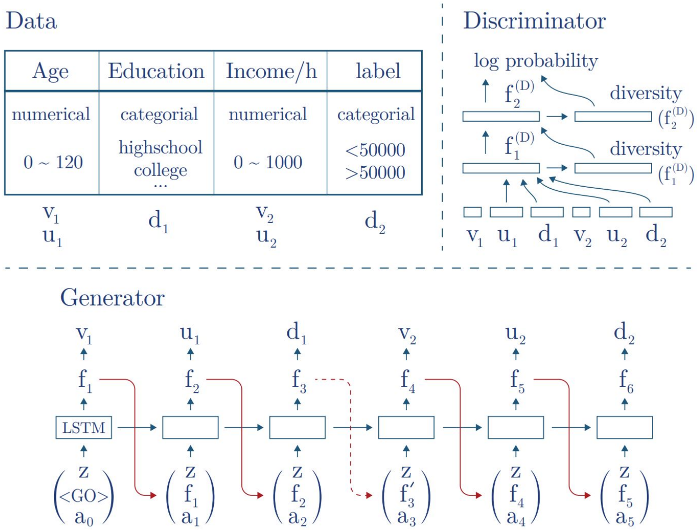
cat	=>	1.2	-0.1	4.3	3.2
mat	=>	0.4	2.5	-0.9	0.5
on	=>	2.1	0.3	0.1	0.4
…			…		

Data Encoding: Nominal values

- Test data might have unseen or higher values than train data.
- Some do it categorical, some continuous.
- Effectiveness depends on the number of unique values. If large number of unique variables, approximate continuously or use embeddings

Related work 1: TGAN[1]

- Encodes continuous values with Gaussian Mixture Models.
- Recurrent Neural Network
- Has an attention mechanism



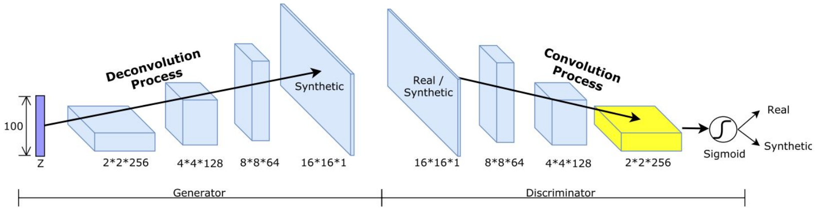
Advantages

- Continuous encoding
- Attention

1 Xu, Lei, and Kalyan Veeramachaneni. "Synthesizing tabular data using generative adversarial networks." arXiv preprint arXiv:1811.11264 (2018).

Related work 2: TableGAN[1]

- Uses DCGAN, SotA method for images
- Encodes everything in range [0,1]



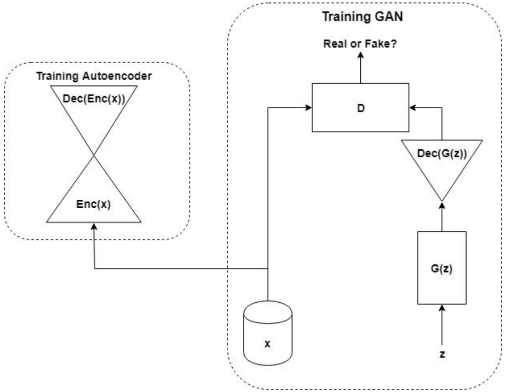
Advantages

- Can capture relations that are quite far apart due to spatial closeness, but also not

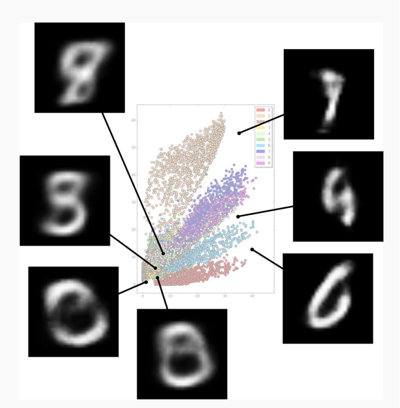
Age 0.25	Working Class 0.1	Education 0.125
Occupation 0.03	Relationship 0.75	0
0	0	0

1 Park, Noseong, et al. "Data synthesis based on generative adversarial networks." Proceedings of the VLDB Endowment 11.10 (2018): 1071-1083.

Related work 3: MedGAN[1]



- Autoencoders translates categorical features to continuous ones
- Circumvent categorical features in the generator
- Sampling autoencoder latent space is arbitrary for locations unseen during training
- Original implementation does not work with multiple data types



Advantages

- No discrete/categorical values in the GAN

1 Park, N., Mohammadi, M., Gorde, K., Jajodia, S., Park, H., & Kim, Y. (2018). Data synthesis based on generative adversarial networks. Proceedings of the VLDB Endowment, 11(10), 1071-1083.

Contributions

Contributions

1. Two improvements on the State of the Art model: TGAN
2. Improved evaluation metric for synthetic tabular data

Proposals

1. Use the WGAN-GP architecture
2. Add skip connections to the generator
3. An aggregate evaluation metric called the Similarity Score

Experimental Setup

1. Take our two GAN versions
2. Compare with three other models
3. Generate 100k rows
4. See how close generated data is to original

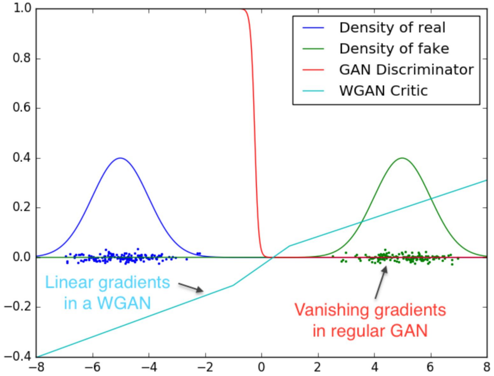
Data

Dataset	#Features	#D	#C	#rows	#labels
Berka	8	**4**	**4**	1056320	3
Census	40	**33**	**7**	199522	2
Creditcard	30	**1**	**29**	248808	2

Proposal 1: WGAN-GP architecture

Built upon TGAN to use the WGAN-GP architecture. This has shown improvements in visual domain with higher image fidelity and improved convergence.

Hypothesis: Model will converge faster, not mode-collapse and yield higher quality samples



Proposal 1: WGAN-GP architecture

1. Adam optimizer parameter change to essentially become RMSProp (momentum=0)
2. Training discriminator more often than generator (5 times)
3. Output of discriminator no longer has sigmoid activation
4. Adapting loss function – use gradient penalty

Classic GAN

$$L_D = \log D(x) + \log(1 - D(G(z))) \rightarrow$$

$$L_G = log(D(G(z)) + \sum_{i=1}^{n_c} KL(u_i', u_i) + \sum_{i=1}^{n_d} KL(\mathbf{d}_i', \mathbf{d}_i) \rightarrow$$

WGAN-GP

$$L_D = \frac{1}{m}\sum_{i=1}^{m} D(x) - \frac{1}{m}\sum_{i=1}^{m} D(G(z))$$

$$L_G = \frac{1}{m}\sum_{i=1}^{m} D(G(z))$$

Proposal 2: skip-connections

- Essential part of many classifiers (ResNets)
- Greater information and gradient retention
- Use it in the Generator

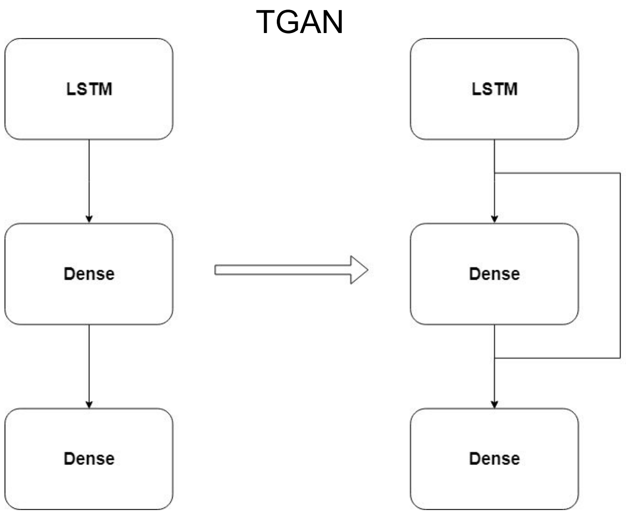
Hypothesis: Model will converge faster and more stable.

Proposal 2: skip-connections

Mathematically:

$$x_k = \mathrm{RELU}(W_k x_{k-1}) + x_{k-1}$$

Visually



Proposal 3: Similarity Score

Aggregate results from several metrics
Given a real and synthetic dataset

1. Correlation coefficient between basic statistical measures (mean, variance, etc.)
2. Correlation coefficient between the correlations of columns within a dataset
3. Correlations between column's distributions between datasets
4. 1 - MAPE of the variance top-5 PCA components
5. Machine learning efficacy
    a. 1 - MAPE for classifiers
    b. Correlation coefficient for RMSE scores

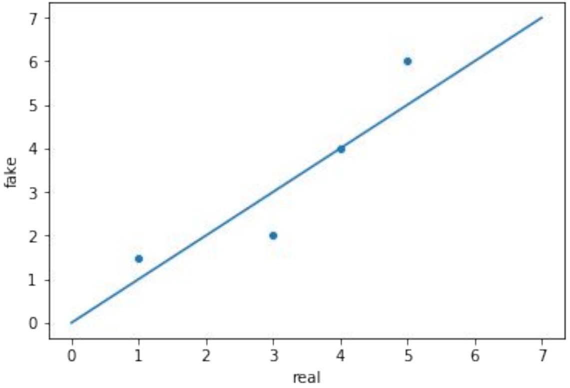
Take the mean for the final Similarity Scores
Give a single value to indicate similarity!

Proposal 3: Similarity Score

1. Basic statistical properties Correlation S_{basic}

Are column distributions consistent?

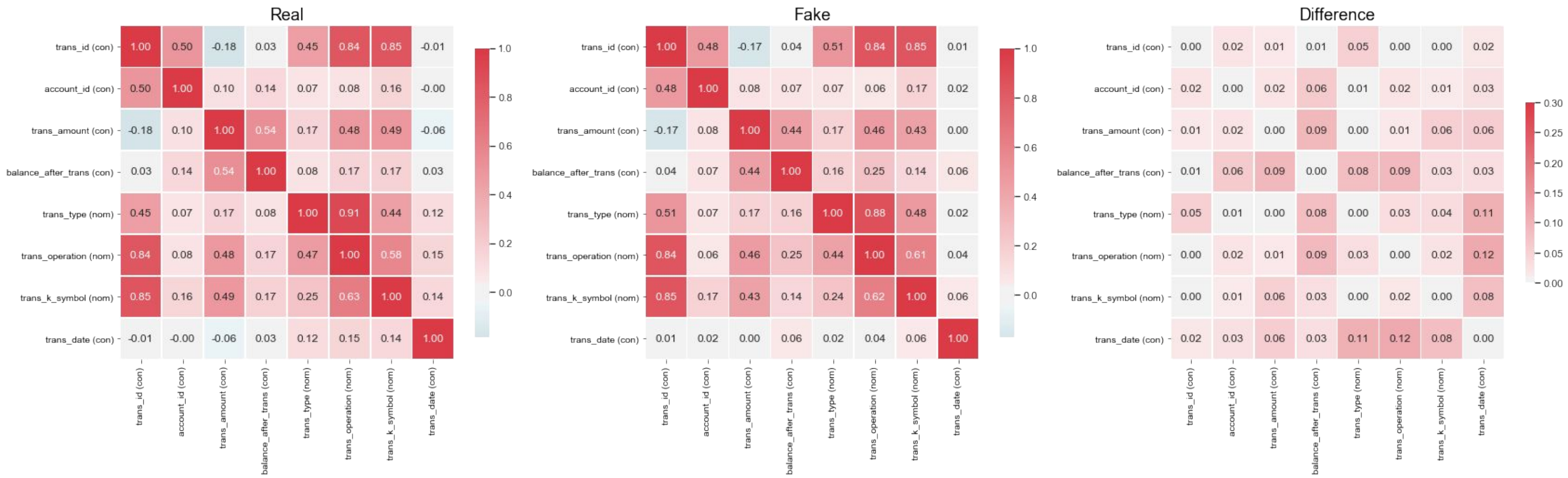
	Real	Fake
Mean Columns 1	5	6
Mean Column 2	3	2
Variance Column 1	4	4
Variance Column 2	1	1,5



Proposal 3: Similarity Score

2. Column Correlations Correlation S_{corr}

Are relations within the datasets consistent?

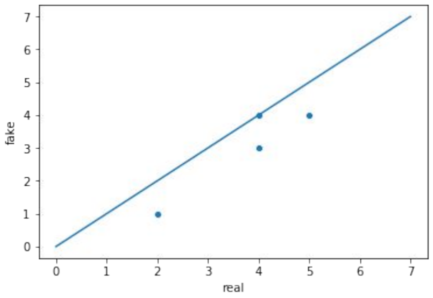


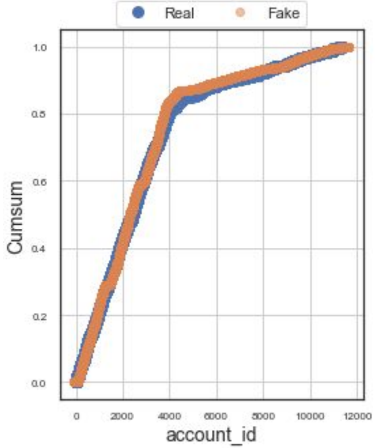
Proposal 3: Similarity Score

3. Mirror column Correlations S_{mirr}

Are relations between the datasets consistent?

Real Column *x*	Fake column *x*
2	1
4	3
4	4
5	4





Proposal 3: Similarity Score

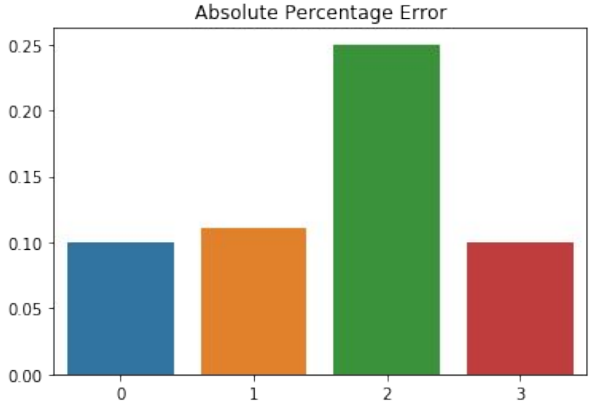
4. PCA Correlations S_{PCA}

Are the PCA explained variances similar?

$$S_{pca} = 1 - MAPE(real\ variance, fake\ variance)$$

Explained Variance

Component	Real	Fake
1	10000	11000
2	450	400
3	8	6
4	0.2	0.22



Proposal 3: Similarity Score

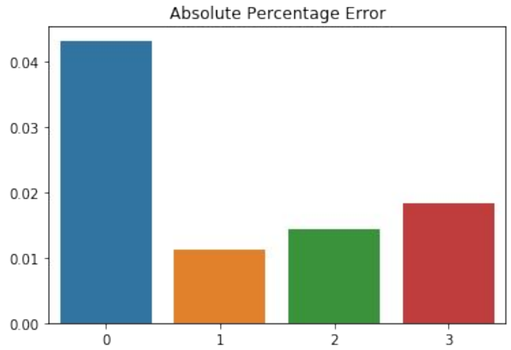
5. Machine learning Efficacy S_{est}

Is the performance on machine learning algorithms comparable?

$$S_{est} = 1 - MAPE(real\ scores, fake\ scores)$$

Estimator scores

Trained on	Model	Real	Fake
Real	Random Forest	0.768	0.735
	Logistic Regression	0.985	0.974
Fake	Random Forest	0.712	0.725
	Logistic Regression	0.923	0.940



Proposal 3: Similarity Score

Average all metrics

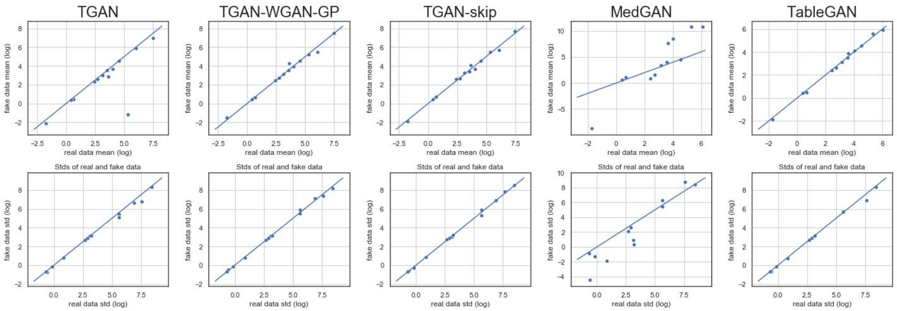
$$SimilarityScore = Average(S_{basic}, S_{corr}, S_{mirr}, S_{pca}, S_{est})$$

Results

Results 1: Basic Statistics Correlations

Correlation coefficient between means, median, std and variance

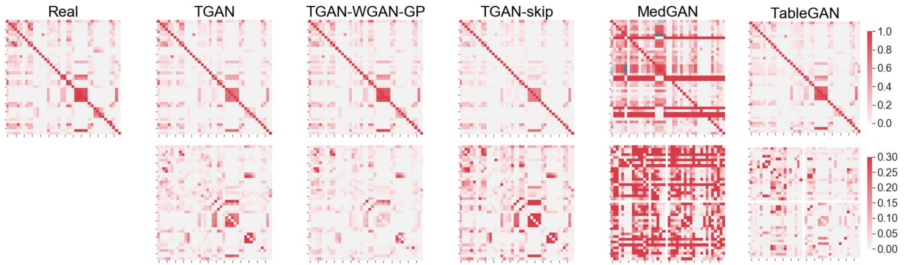
		Best			
Dataset	**TGAN**	**WGAN**	**SKIP**	**MedGAN**	**TableGAN**
Berka	0.9910	**0.9955**	0.9850	0.9115	0.9895
Census	0.9212	**0.9909**	*0.9894*	0.4325	*0.9947*
Creditcard	0.8028	*0.8661*	***0.8799***	-0.0329	*0.8734*



Results 2: Column Correlations

Correlation coefficient between column correlations

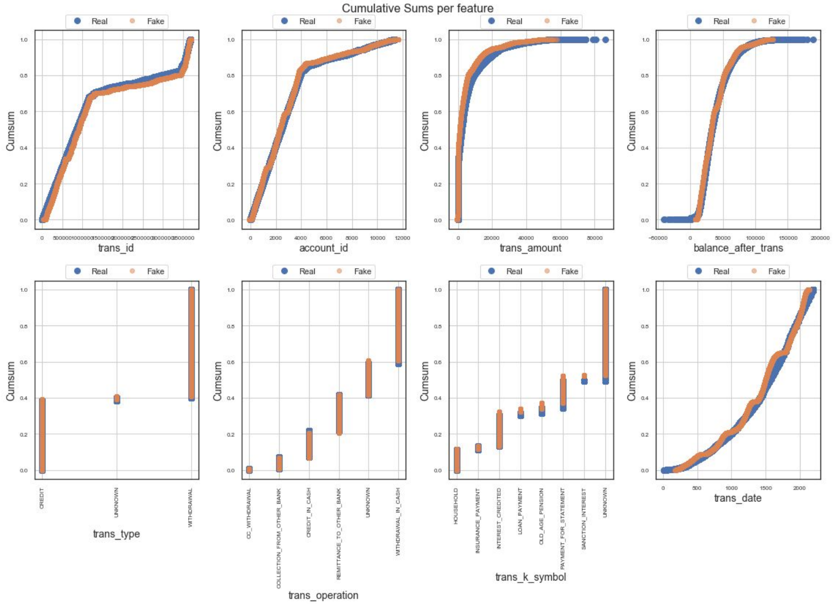
Dataset	TGAN	WGAN (Best)	SKIP	MedGAN	TableGAN
Berka	0.9821	0.9470	**0.9832**	0.7694	0.6468
Census	0.9581	**0.9773**	*0.9053*	0.0644	*0.9128*
Creditcard	0.0968	**0.2932**	*0.2114*	-0.0471	*0.2157*



Results 3: Mirror Column Correlations

Correlation coefficient between datasets

Dataset	TGAN	WGAN (Best)	SKIP	MedGAN	TableGAN
Berka	0.9276	0.9150	**0.9572**	0.5602	0.8864
Census	0.7008	**0.8722**	*0.7941*	0.2092	*0.8651*
Creditcard	0.9215	**0.9605**	*0.9342*	0.7888	*0.9425*



Results 4: PCA Variance Correlation

1 – MAPE(Real Variance, Fake Variance)

Dataset	Best TGAN	WGAN	SKIP	MedGAN	TableGAN
Berka	**0.9456**	0.9399	0.9424	0.8236	0.9465
Census	0.9507	**0.9739**	*0.9643*	0.6907	*0.9748*
Creditcard	**0.8501**	0.7810	0.7266	0.1726	0.8667

Results 5: Estimator results

1 – MAPE(real F1/real RMSE, fake F1/fake RMSE)

Dataset	TGAN	WGAN (Best)	SKIP	MedGAN	TableGAN
Berka	**0.9929**	0.9807	0.9572	0.4168	0.8899
Census	0.9854	**0.9871**	0.9840	-292.1544	*0.9673*
Creditcard	0.7999	**0.9817**	*0.9712*	-0.8533	0.1696

Results: Similarity scores

Dataset	TGAN	Best WGAN	SKIP	MedGAN	TableGAN
Berka	**0.9678**	0.9556	0.9464	0.6963	0.8718
Census	0.9032	**0.9603**	*0.9274*	-58.1515	*0.9429*
Creditcard	0.6942	**0.7765**	*0.7447*	0.0056	0.6136

Takeaways

Models

1. Using the WGAN-GP architecture typically better than classic GAN
2. Skip connections typically improve on non-skip variant
3. Combining both would likely work very well. Coincidentally, this is exactly what happened in a follow-up paper from MIT[1]

Evaluation

1. Similarity score gives consistent single value performance indicator
2. Can be split into its parts for detailed information

1. Xu, L., Skoularidou, M., Cuesta-Infante, A., & Veeramachaneni, K. (2019). Modeling Tabular data using Conditional GAN. In *Advances in Neural Information Processing Systems* (pp. 7333-7343).

Thanks you.
Questions?