Modeling Tabular Data using Conditional GAN

Lei Xu[1], Maria Skoularidou[2], Alfredo Cuesta-Infante[3], Kalyan Veeramachaneni[1]
[1]MIT LIDS, [2]University of Cambridge, [3]Universidad Rey Juan Carlos

Introduction

- GANs can generate realistic synthetic images. But in generating synthetic tabular data, state-of-the-art GAN-based models cannot outperform simple Bayesian network models as shown on Table 1.
- The challenges of generating synthetic data using GANs are the non-Gaussian multimodal distribution of continuous columns and imbalanced discrete columns.
- We design CTGAN to address these challenges. CTGAN uses *mode-specific normalization* to effectively represent continuous values from different distribution; and uses a *conditional generator* and a *training-by-sampling* method to learn imbalanced discrete columns.

Table 1: The number of wins: Deep learning vs. Bayesian Networks on 8 real datasets.

	outperforms	
Method	CLBN	PrivBN
MedGAN, 2017	1	1
VeeGAN, 2017	0	2
TableGAN, 2018	3	3
CTGAN	**7**	**8**

Mode-specific Normalization

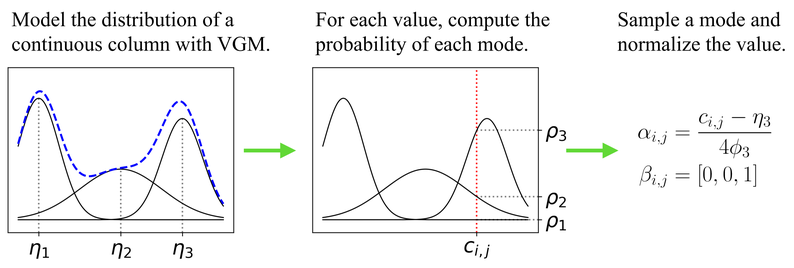

Figure 1: Mode-sepcific normalization can infer the number of modes in a continous column, then represent the column as a scalar value in range $[-1, 1]$ and a one-hot vector.

Conditional Generator

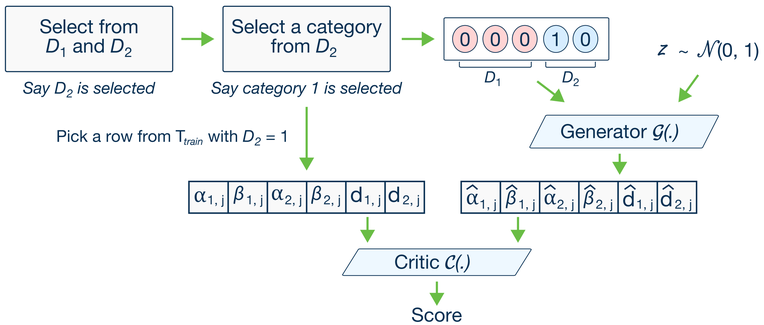

Figure 2: *Conditional generator* and *training-by-sampling* in CTGAN model.

Evaluation Metrics

- For simulated data, we evaluate (1) the likelihood of test data on learned distribution as $\mathcal{L}_{test}$, (2) and the likelihood of synthetic data on original data distribution as $\mathcal{L}_{syn}$.
- For real data, we train classifiers or regressors on the synthetic data and evaluate prediction metrics on real test data.

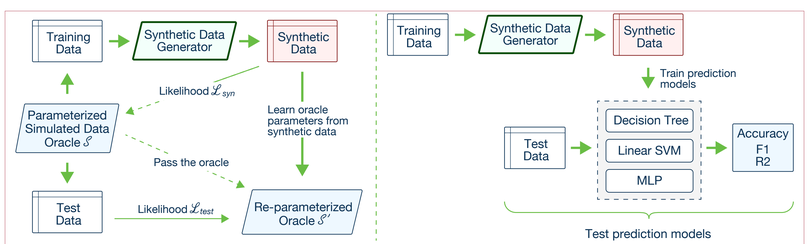

Figure 3: Evaluating efficacy of synthetic data.

Experiments

Datasets:
3 Gaussian mixture datasets.
5 Bayesian network datasets.
8 Real datasets.

Results
Our CTGAN model outperforms other BN-based and GAN-based models on Gaussian mixture datasets and real datasets.

Table 2: Benchmark results.

	GM Sim.		BN Sim.		Real	
Method	$\mathcal{L}_{syn}$	$\mathcal{L}_{test}$	$\mathcal{L}_{syn}$	$\mathcal{L}_{test}$	clf	reg
Identity	-2.61	-2.61	-9.33	-9.36	0.743	0.14
CLBN	**-3.06**	-7.31	**-10.66**	**-9.92**	0.382	-6.28
TableGAN	-8.24	-4.12	-11.84	-10.47	0.162	-3.09
CTGAN	-5.72	**-3.40**	-11.67	-10.60	**0.469**	**-0.43**

Ablation Study:

- We replace mode-specific normalization with a simple min-max normalization. The performance drops $\mathbf{25.7\%}$.
- We disable the *training-by-sampling* method, the performance decreases $\mathbf{17.8\%}$.
- We disable the *conditional generator* as well as *training-by-sampling*, the performance descreases $\mathbf{36.5\%}$.

Conclusion

In this paper we attempt to find a flexible and robust model to learn the distribution of columns with complicated distributions. We observe that none of the existing deep generative models can outperform Bayesian networks which discretize continuous values and learn greedily. We show that our model can learn better distributions than Bayesian networks. Mode-specific normalization can convert continuous values of arbitrary range and complicated distribution into a bounded vector representation suitable for neural networks. And our *conditional generator* and *training-by-sampling* can overcome the imbalanced training data issue. As future work, we would derive a theoretical justification on why GANs can work on a distribution with both discrete and continuous data.