GAN Applications with Discrete Data

Reproducing and Bringing State-of-the-Art to Industry

Ramiro CAMINO

SEDAN Lab, SnT, University of Luxembourg, Luxembourg

ramiro.camino@uni.lu — (+352) 46 66 44 - 6855

Abstract

Gaining insight into financial transactions is a legal requirement for banks and financial service providers. Unfortunately, for most data available to banks and financial service provider, the truth about the transaction purpose is unknown and illicit activities have a low base rate. Additionally, after reporting suspicious activity, banks will no hearing back from investigations. Consequently, many supervised approaches, which inherently rely on the ground truth, cannot be used. In the following we study the use of Generative Adversarial Networks (GAN) for data augmentation to increase the scarce amount of known fraudulent cases, and for data imputation to improve the quality of samples.

Introduction

- Issues analyzing financial transactions and accounts:
  - Class imbalance: e.g. fraud small / non-fraud big
  - Unavailability and unreliability of labels
  - Missing values along multiple features
  - Impossibility to share private data
- Possible applications of GANs:
  - Data augmentation for the minority class
  - Data imputation for missing values
  - Synthetic data generation to replace private data
  - Unsupervised feature extraction
- State-of-the-Art for GANs:
  - Plenty of work and promising results for continuous domains like images, video and sound
  - Studies involving discrete samples are mainly focused on sequences of words, characters or symbols

Related Work

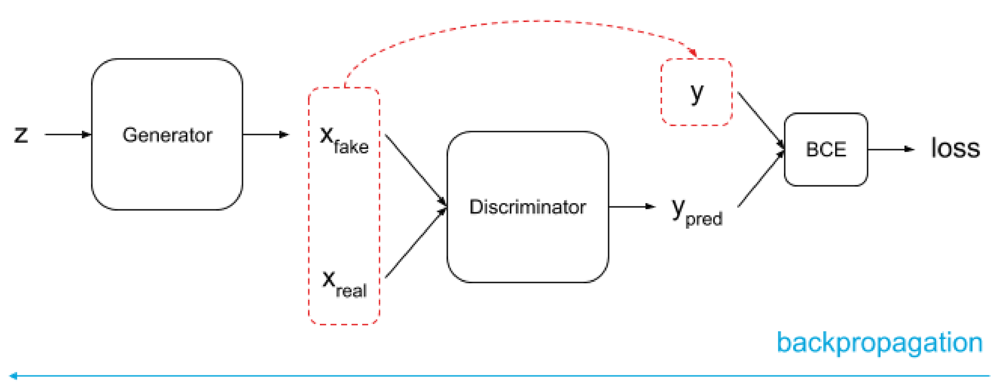

Figure 1: Generative Adversarial Network (GAN) [4]. The generator weights can be trained thanks to the error backpropagated from the discriminator loss and through the discriminator weights.

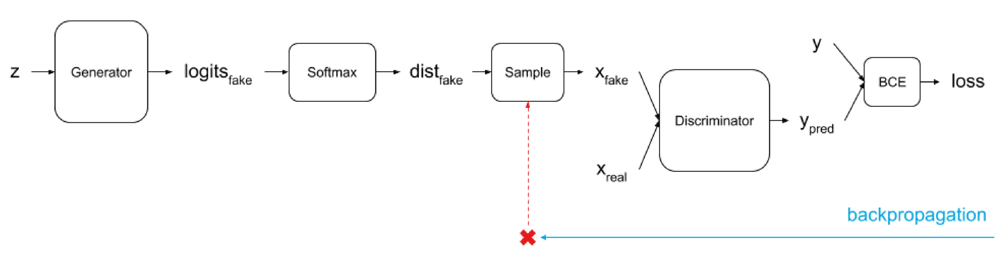

Figure 2: When sampling from a discrete distribution, the backpropagation fails because the sample operation is non-differentiable.

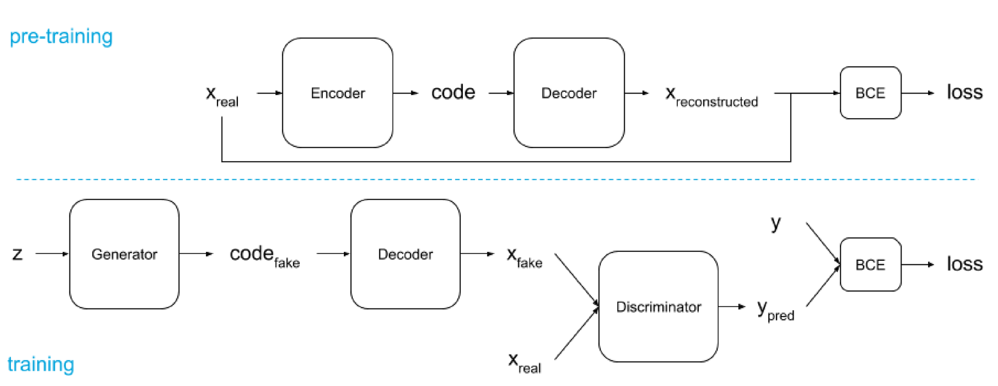

Figure 3: MedGAN [2] first pre-trains an autoencoder. During the second training phase, the decoder transforms the generator outputs into samples where each feature is considered as a binary variable.

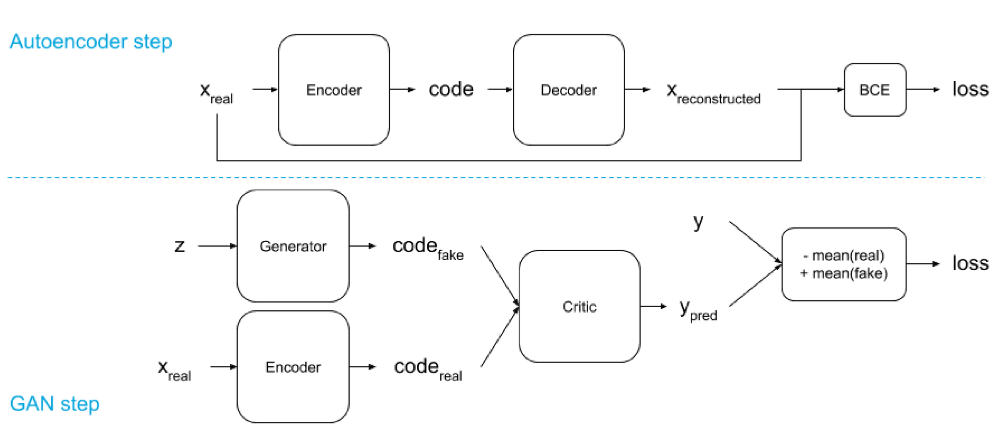

Figure 4: ARAE [7] trains and autoencoder and a WGAN [1] with alternating steps. The discriminator is replaced by a critic (along with the according loss), and takes as input codes instead of samples.

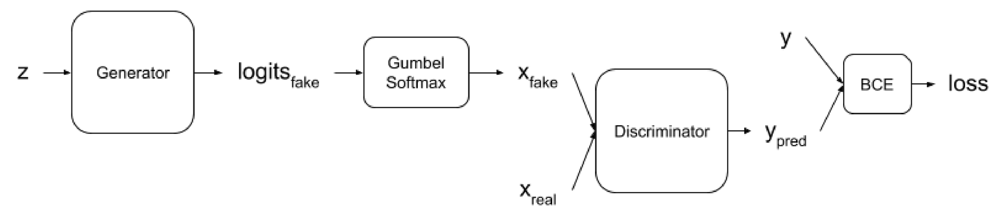

Figure 5: The Gumbel-Softmax [6] and the Concrete-Distribution [10] were simultaneously proposed to sample from discrete distributions in the domain of VAEs [8]. Later [9] adapted the technique to GANs for sequences of discrete elements as shown in this diagram.

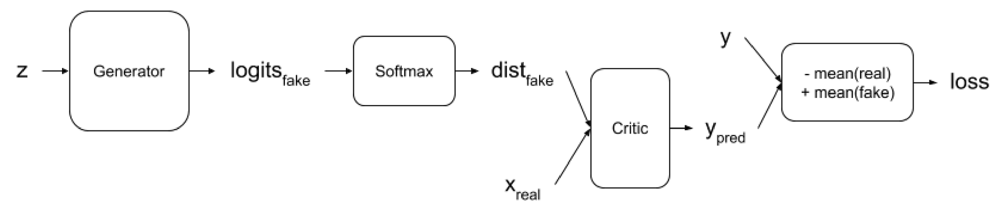

Figure 6: In [5] authors propose to add a gradient penalty to the WGAN loss (hence the name WGAN-GP) to improve training. Additionaly, the authors claim that with WGAN-GP it is possible to generate discrete sequences training directly with softmax outputs.

Multi-Categorical GAN

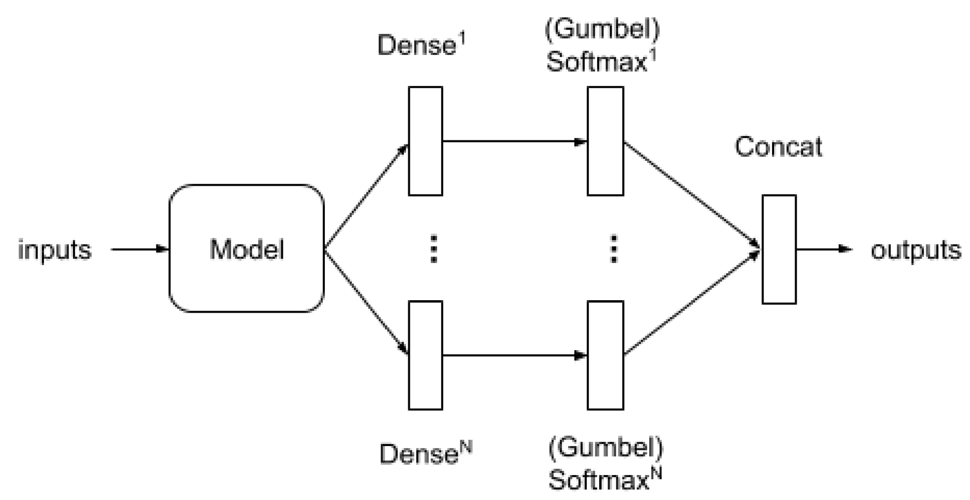

Figure 7: Proposed architecture for either generator or decoder models. After the model output, we place in parallel a dense layer per categorical variable, followed by an activation and a concatenation to obtain the final output. The choice between Gumbel-Softmax or Softmax depends on the model. The reconstruction losses should be transformed into the sum of one cross entropy loss per variable.

Experiments

- Split dataset in $\mathcal{D}_{train}$ and $\mathcal{D}_{test}$
- Train generative model with $\mathcal{D}_{train}$
- Generate $\mathcal{D}_{sample}$ with the trained model
- Experiments proposed in [2]:
  - Probabilities by dimension: compare proportion of ones per dimension between $\mathcal{D}_{test}$ and $\mathcal{D}_{sample}$
  - Predictions by dimension: train one predictive model per dimension with $\mathcal{D}_{train}$ and evaluate with $\mathcal{D}_{test}$; repeat but training $\mathcal{D}_{sample}$ and compare results.
- Predictions by categorical: analogous experiment but training one predictive model per categorical variable.
- Example results shown in Figure 8

Conclusions

- Compared to unmodified MedGAN and ARAE, all approaches improve the performance across all datasets
- Even though we cannot identify a clear best model
- The performance improvement comes at the cost of requiring to add extra information (the variable sizes). In some cases, this information may not be available.

Future Work

- Analyze datasets with mixed variable types
- Explore more hyperparameters and architecture options
- Experiment with oversampling and undersampling
- Improve validation with domain specific metrics

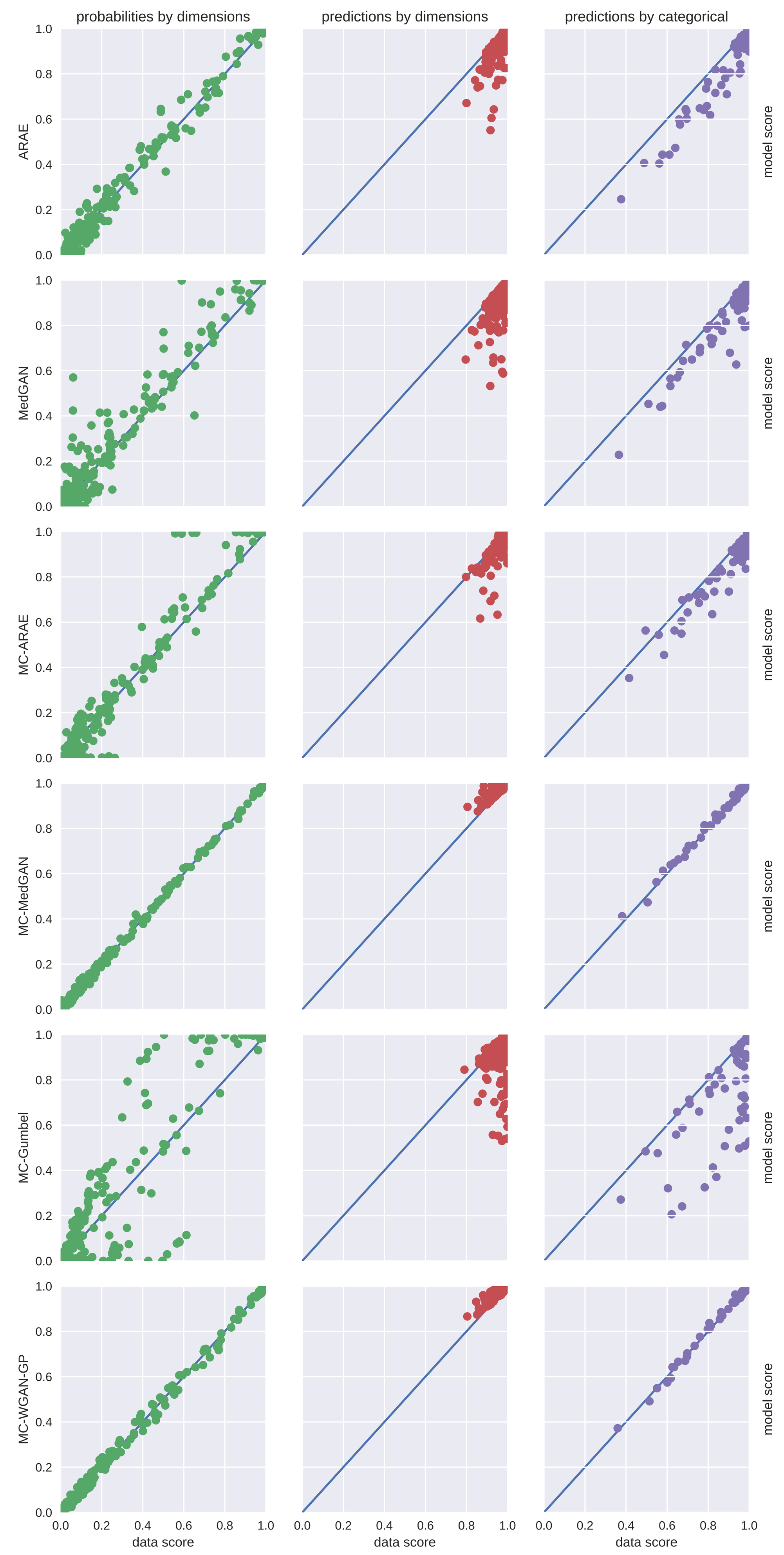

Figure 8: Experiments for the US Census 1990 [3]. Each column corresponds to a metric and each row a to sample from a different model.

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein GAN. *arXiv preprint arXiv:1701.07875*, 2017.

[2] Edward Choi, Siddharth Biswal, Bradley Malin, Jon Duke, Walter F. Stewart, and Jimeng Sun. Generating multi-label discrete patient records using generative adversarial networks. In *PMLR*, pages 286–305.

[3] D. Dua and E. Karra Taniskidou. UCI Machine Learning Repository. 2017.

[4] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc.

[5] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein GANs.

[6] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax.

[7] Junbo, Zhao, Yoon Kim, Kelly Zhang, Alexander M. Rush, and Yann LeCun. Adversarially regularized autoencoders.

[8] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes.

[9] Matt J. Kusner and José Miguel Hernández-Lobato. GANS for sequences of discrete elements with the gumbel-softmax distribution.

[10] Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables.