A fast and high fidelity non-Deep Learning synthetic data generator for tabular data

Max Baak
ING Bank
Wholesale Banking Advanced Analytics (WBAA)
Bijlmerdreef 106
1102 CT Amsterdam
Netherlands
Max.Baak@ing.com

Abstract

Current state-of-the-art approaches towards generating synthetic versions of real, tabular, approximately i.i.d. data by and large make use of the immense progress and representational power of Generative Adversarial Networks and Variational AutoEncoders. We present an approach to the problem which generates results comparable or superior to the existing approaches, with a computational cost an order of magnitude (&&& CHECK THIS FROM SIMON'S RESULTS) lower.

1 Introduction

One of the current bottlenecks to further progress in deployment of machine learning applications are the issues surrounding protecting the privacy of sensitive data, and also respecting data license agreements which may prohibit the duplication of a licensed dataset. These barriers manifest as delays -even within a given institution- in providing access to data to the Data Scientists who require it, or even being able to create a training set when the data licensing requirements constrain use to a single copy of the data, such that this data can be used in production only, and surrogate, synthetic data is required to train in development, testing, and acceptance environments of the DTAP testing and deployment approach.[1] A major stumbling block in fostering further collaboration between industry and academia, or indeed with the wider Data Science community is often the inability to share data due to privacy issues, which is something reliable synthetic data generation could alleviate.

In order to safely enable the use of realistic but entirely synthetic datasets in these scenarios, reliable tools to model and sample from estimated forms of the empirical joint distributions of the data are required. It is towards this goal that the present work makes progress, with focus on tabular, numeric, approximately independent and identically distributed data.

In the subsequent sections we detail a novel method to learn and generate synthetic data from an existing original dataset, where use of Deep Learning or other high complexity models is entirely optional. The computational cost of fitting is therefore significantly reduced, but without hampering the fidelity of the generated samples, as judged by multiple performance metrics. In section 1.1 an overview of prior work related to the present work is overviewed, in section 2 we detail the method used, and section 3 presents the results obtained. Potential paths for further work are outlined in section 4.

[1] The even more ambitious use case of using synthetic data generators to mitigate overfitting by avoiding data reuse and training on endless synthetically produced training sets remains elusive until synthetic data generation tools are deemed reliable enough to warrant the leap of faith of treating them as being truly almost as good as the true underlying data generation mechanism.

Preprint. Under review.

POSSIBLY A SENTENCE ON THE USE CASE WITHIN HEP WHICH FIRST INITIATED THIS WORK, MAX? NOT REQUIRED OR STRICTLY RELEVANT, BUT MIGHT BE OF INTEREST TO THE READER IN UNDERSTANDING THE DISTINCT APPROACH TAKEN.

1.1 Prior work

Efforts to produce realistic synthetic datasets span a wide array of methods, from re-sampling techniques and perturbations applied to the original dataset, to datasets generated from manually curated rules. For a review of some of the existing non-Deep Learning approaches see Surendra and Mohan [2017].

In recent years however, encouraged perhaps by the astounding success of GANs and VAEs in successfully generating sythetic images which are fully sampled from their learned weights and yet virtually indistinguishable from real images (for GANs see Karras et al. [2020] and for VAEs see Razavi et al. [2019]), approaches of this type have been created for tabular data, with a benchmark emerging for the evaluation and comparison of these methods Xu et al. [2019].

In particular, focused on potential applications within finance, of the type which motivated the present work, the NeurIPS 2019 conference included an overview of the field and some of its challenges Assefa et al. [2020].

2 Method

The method broadly consists of two stages: learning and removal of linear and distributional characteristics of the data, followed by learning of the residual non-uniform associations in the data. Once these two stages have been fit, the process can then be reversed, generating non-uniform associations as learnt from the data, and regenerating the linear and distributional characteristics of the data, incorporating the necessary Jacobian factors as needed, and thereby effectively sampling from the learned joint distribution.

The code for our method is available at: `https://github.com/mbaak/synthsonic`.

2.1 Quantile transformation to Normality

Describe Quantile transformation to Normality, motivated by the fact that PCA effectively assumes Normal noise Tipping and Bishop [1999]

2.2 Rotation of linear features

PCA used to rotate-away and learn linear features

2.3 Quantile transformation to uniformity

Any remaining distributional shape is fitted and removed

2.4 Fitting of residuals

A classifier is trained to distinguish between residuals remaining after the procedure above, and uniform datapoints. This classifier is ensured to be well-calibrated, describe how

2.5 Closed form fitted probability density function

Derivation of Jacobian required and final form of pdf

2.6 Treatment of categoricals

Describe how those were dealt with here

3 Results

We evaluated the method with respect to multiple criteria: with regards to a likelihood criteria as set out in the SDGym benchmark and leaderboard in Xu et al. [2019], as well as ensuring marginals of the learnt distributions matched the original datasets, multiple modes -if existent- were preserved, and lastly also in an adverserial way, judged by a discriminating model's ability to distinguish between the synthetic and the original dataset (TO DO).

Lastly, in order to mitigate the risk of having overfit to the SDGym datasets, we also evaluated its performance on synthetic datasets which we manually created, with well known design features, to test the method's ability to recover these and therefore generalise beyond the datasets it was initially applied to.

3.1 The SDGym benchmark

We beat all the approaches in Xu et al. [2019] (VERIFY THIS)

3.2 Marginals

Plots of marginals showing overlap between real and fake datasets Show at least one multi-modal fit as well

3.3 Adversarial validation

4 Outlook

Improve treatment of categoricals? Further improvements on best choice of classifier Anything else

SECTIONS BELOW STILL TO BE FILLED OUT, BUT ARE PRO FORMA

Broader Impact

Authors are required to include a statement of the broader impact of their work, including its ethical aspects and future societal consequences. Authors should discuss both positive and negative outcomes, if any. For instance, authors should discuss a) who may benefit from this research, b) who may be put at disadvantage from this research, c) what are the consequences of failure of the system, and d) whether the task/method leverages biases in the data. If authors believe this is not applicable to them, authors can simply state this.

Use unnumbered first level headings for this section, which should go at the end of the paper. **Note that this section does not count towards the eight pages of content that are allowed.**

Acknowledgments and Disclosure of Funding

Use unnumbered first level headings for the acknowledgments. All acknowledgments go at the end of the paper before the list of references. Moreover, you are required to declare funding (financial activities supporting the submitted work) and competing interests (related financial activities outside the submitted work). More information about this disclosure can be found at: `https://neurips.cc/Conferences/2020/PaperInformation/FundingDisclosure`.

Do **not** include this section in the anonymized submission, only in the final paper. You can use the `ack` environment provided in the style file to autmoatically hide this section in the anonymized submission.

References

References follow the acknowledgments. Use unnumbered first-level heading for the references. Any choice of citation style is acceptable as long as you are consistent. It is permissible to reduce the

font size to `small` (9 point) when listing the references. **Note that the Reference section does not count towards the eight pages of content that are allowed.**

References

HMHS Surendra and HS Mohan. A review of synthetic data generation methods for privacy preserving data publishing. *International Journal of Scientific & Technology Research*, 6(3):95–101, 2017.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119, 2020.

Ali Razavi, Aaron van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. In *Advances in Neural Information Processing Systems*, pages 14866–14876, 2019.

Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. Modeling tabular data using conditional gan. In *Advances in Neural Information Processing Systems*, pages 7335–7345, 2019.

Samuel Assefa, Danial Dervovic, Mahmoud Mahfouz, Tucker Balch, Prashant Reddy, and Manuela Veloso. Generating synthetic data in finance: opportunities, challenges and pitfalls. *Challenges and Pitfalls (June 23, 2020)*, 2020.

Michael E Tipping and Christopher M Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, 1999.