Generating Synthetic but Plausible Healthcare Record Datasets

Laura Aviñó[+*], Matteo Ruffini[*], Ricard Gavaldà[*†]

[+]Esci-UPF Barcelona, Spain; [*] UPC Barcelona, Spain; [†]BGSMath, Barcelona, Spain

Why?

The need for synthetic datasets that “look like” realistic cases is clear in many fields of science and engineering.
For instance:

- Hospitals and companies may want to share their data, but they cannot due to privacy issues
- Sometimes methods need more data than the ones available

A solution of this is generating synthetic data that looks similar.
This is especially important for health records since the privacy is important.

In recent years, machine learning research on generative models has been boosted by the success of Generative Adversarial Networks (GANs). However, they have some problems:

- Hyper-parametrization
- Poor interpretation
- Collapse mode

What?

We developed a model-based approach that assumes data to be generated by a certain latent variable model – precisely, a Naive Bayes model with binary features – which is learned using the method of moments in [1] used to cluster patients with similar clinical profiles; here, we leverage on the generative nature of the considered Naive Bayes model, using it to sample
realistic synthetic data. The avantatges:

- It is faster to set up and run than Gans as there is only one parameter to tune (the number of clusters)
- Ideally generates not only realistic instances but populations.
- Easy to interpret

.

How?

Background

Let assume a data st D from by N rows or intances (patients) and d columns or features (diagnostics).
We also assume that the model is generated by a Naive Bayes model. It has:

- A latent (non observable) variable Y from a discrete distribution. The true clinical status
- A vector $X = (X_1 .. X_d)$ of binary observable varibles. The variables $X_1,...,X_d$ are conditionally independent given Y.

We can think about them as the real manifestation of the patient status given her/his real clinical status.

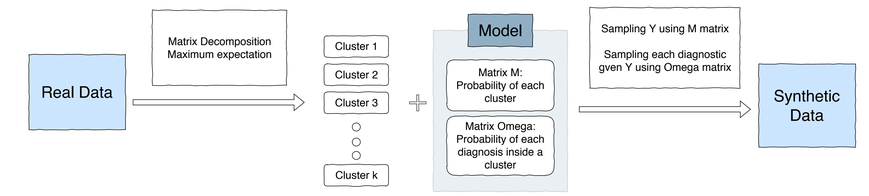
DataSets

We will consider two datasets.

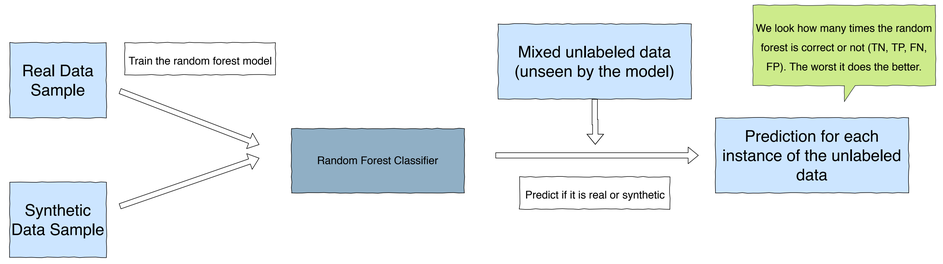
- One will be MIMIC III [2] a publicly available dataset containing medical data from the Beth Israel Deaconess Medical Center regarding the years between 2001 and 2012.
- The second dataset was provided by Hospital de la Santa Creu i Sant Pau in Barcelona1.

In particular, we will focus on the diagnostic ICD9 records, a sub-dataset whose rows represent the visits of patients to the hospital, and the columns contain the codes of the diagnostics annotated by the doctors (1 if so, 0 if not).

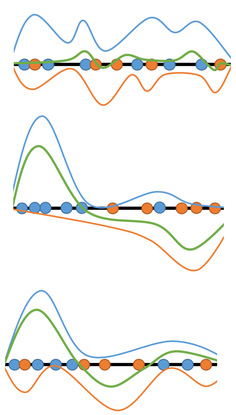
Generation of data



Computing performance



Test by similarity of the population: Mean Maximum Discrepancy [3]



Let's assume we have two populations, the blue one and the orange one. The blue distribution generates the blue population and the orange distribution generates the orange population, note that they have opposite signs. Then, we subtract the orange distribution to the blue one. This difference is represented by the green distribution. How hard is for the green distribution to generate both populations give us an intuition of the MMD value.

- In the first figure, both populations are similar, so, distributions are annulled. Then, the probability for the green distribution will be low.
- In the second figure, populations are distant. Then, the green distribution is positive where there are blue dots and negative where there are orange dots. So, the MMD value would be high.
- Finally, we see a collapse mode scenario. If one, for instance, the blue population is concentrated in a place, the green distribution will have that sign. So, it will easily generate instances for that distribution but it would not generate dots for the other distribution.

Results

For each data set, we generated synthetic data using MedGan [4] (a GAN approach), a baseline (generation of patients only using the frequencies of each diagnosis) and our method with a different number of clusters. Then, for each synthetic sample, we computed the MMD, the lower the better, and the accuracy, recall, precision and specificity of the Random Forest predictor. Unlike most ML papers, worse predictor performance is better for our purposes (harder to identify synthetic datasets). The code for generating the results for Mimic is in https://github.com/LauAvinyo/tensorGen.

Table 1: MIMIC

	Accuracy	Recall	Precision	Specificity	MMD
Baseline	0.86	0.84	0.88	0.89	0.12
MedGan	0.82	0.75	0.88	0.90	0.50
5 clusters	0.74	0.69	0.77	0.80	0.04
10 clusters	0.69	0.63	0.72	0.76	0.05
100 clusters	0.59	0.52	0.60	0.65	0.01

Table 2: Congestive Heart Failure, Sant Pau

	Accuracy	Recall	Precision	Specificity	MMD
Baseline	0.72	0.68	0.75	0.77	0.59
MedGan	0.67	0.58	0.70	0.75	3.92
5 clusters	0.65	0.60	0.67	0.70	0.20
10 clusters	0.61	0.55	0.61	0.67	0.09
100 clusters	0.53	0.45	0.53	0.61	-0.01

As can be seen, MedGan performs better than the baseline in the usual metrics, but its MMD is actually quite worse than the baseline; we attribute this to the effect of the mode collapse. Our method performs better than both even if using only 5 latent values (clusters), and gets remarkably low MMD even then. Values improve noticeably up to 100 clusters.
Table 2 shows the results for the second dataset. Again, MedGan does far worse than the baseline in MMD, our method is better on all measures at 10 clusters, and remarkably good at 100 clusters. Another aspect to remark is computation time. The experiments with MedGan while our method with k = 10 clusters takes a few minutes. For k = 100 the times are in the same order, but one should take into account that MedGan is using the GPU while our method does not. Furthermore, experiments for MedGan have to repeated for a much larger set of hyperparameter value if one wants to be reasonably sure
that “good” values have been found.

Conclusions and future work

Our method has proven to outperform the GAN-basedMedGan on two patient datasets, achieving remarkably low values of MMD and being much harder to distinguish from the real dataset. Besides experimenting on other datasets, future work could include

- Investigate in more depth whether the difference in performance of the tested method can be attributed to mode
  collapse events.
- Parallelizing our method to take advantage of GPU.
- At the theoretical level, investigating whether hard privacy claims can be made about the result of our method, for example in the framework of differential privacy

Anknowledgements

We are greatful to the MIMIC project and Hospital de Sant Pau for providing the data, and to Drs. Julianna Ribera, Salvador Benito, and Mireia Puig for their advice. Research partially supported by grants TIN2017-89244-R and MDM-20140445 from MINECO (Ministerio de Economia, Industria y Competitividad) and the recognition 2017SGR-856 (MACDA)
from AGAUR (Generalitat de Catalunya)

References

[1]: Matteo Ruffini, Ricard Gavald`a, and Esther Limon. 2017. Clustering Patients with Tensor Decomposition. In Proceedings ofthe Machine Learning for Health Care, MLHC 2017, Boston, Massachusetts, USA, 1819August 2017. 126–146. http://proceedings.mlr.press/v68/ruffini17a.html
[2]: Alistair EW Johnson, Tom J Pollard, Lu Shen, H Lehman Li-wei, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. 2016. MIMIC-III, a
freely accessible critical care database. Scientific data 3 (2016), 160035
[3]: Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Sch¨olkopf, and Alexander Smola. 2012. A kernel two-sample test.
Journal of Machine Learning Research 13, Mar (2012), 723–773.
[4]: Edward Choi, Siddharth Biswal, Bradley Malin, Jon Duke, Walter F. Stewart, and Jimeng Sun. 2017. Generating Multi-label Discrete Electronic Health Records using Generative Adversarial Networks. CoRR
abs/1703.06490 (2017). http://arxiv.org/abs/1703.06490