Generative Adversarial Networks and Graphical Structure Learning

Elisa Dowling
Department of Statistics
Rice University
dowling@rice.edu

Emily Wang
Department of Statistics
Rice University
emilywang@rice.edu

Abstract

As a way of addressing privacy concerns, generative adversarial networks (GANs) are specifically designed to produce synthetic data - artificially generated data which possess the statistical properties of the original dataset. These algorithms have recently become popular due to their effectiveness and high accuracy in learning the distribution of the original data. However, the viability of using GAN-produced synthetic data in place of the original data to develop various types of mathematical models is not well explored. In this paper, we investigate the potential differences that may arise when applying complex models to a purely synthetic dataset. Specifically, applying graphical structure learning methods tests the depth of the statistical similarity between synthetic data produced by GANs and the original data. We first apply simple stepwise forward model selection, then birth-death MCMC for Gaussian graphical models, using both the synthetic and original data. Comparing results, we conclude that although GANs may produce a synthetic dataset that statistically resembles the original dataset at first glance, results from applying graphical structure learning algorithms on the two datasets are in fact substantially different.

1 Introduction

With a growing need to protect sensitive data, the importance of developing viable synthetic datasets becomes more and more apparent. In recent years, several methods [6], such as synthetic minority over-sampling [1], differentially private data synthesizer [3], and most recently the generative adversarial network [2], have been introduced to solve such a problem. Though such methods are proven to excel at producing data that is nearly impossible to distinguish from the original data at first glance, the effectiveness of using synthetic data in place of the original data to fit mathematical models is uncertain.

Theoretically, fitting models from synthetic data should yield similar, if not identical, results to the original data as they both should possess the same statistical properties. In this paper, we evaluate this claim by first creating synthetic data using a generative adversarial network and then applying graphical model selection methods on both the synthetic and original datasets. The focus will be on discovering whether synthetic data and original data result in similar graphical model selection, while generative adversarial networks' efficiency in privatizing data is left as future work.

The outline of this paper is as follows. In section 2, we introduce the diagnostic breast cancer dataset. In section 3, we introduce generative adversarial networks as a method of creating synthetic data. In section 4, we compare the results from two graph structure learning models on both the synthetic and original data. In section 5, we conclude our findings and delve into several possibilities for future work.

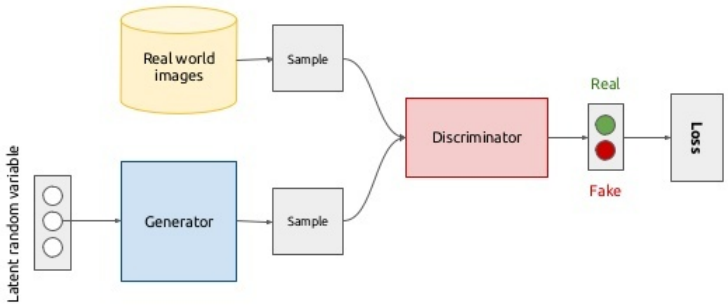

Figure 1: GAN Mechanism depicting opposing generator and discriminator networks

2 Data Set

For our analysis, we used the Wisconsin diagnostic breast cancer dataset from the UCI Machine Learning repository [7]. This dataset is comprised of 699 observations of cell nuclei collected from a fine needle aspiration biopsy of breast masses. Each observation has 13 variables recorded including a unique ID number, diagnosis, and ten additional physical features including radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, and fractal dimension. Approximately 60% of samples were diagnosed benign, and the remainder were diagnosed malignant. This data set was chosen to be used as a point of comparison since there are strong relationships between the physical features of the nuclei and ultimate diagnosis.

3 Generating Private Data

For generating synthetic data, we chose the generative adversarial network, due to its rapidly growing popularity, relatively short computation times, and ability to handle high-dimensional data.

3.1 Generative Adversarial Networks

Generative adversarial networks (GANs) are a family of unsupervised machine learning algorithms, composed of two neural networks that compete with each other in a zero-sum game (Figure 1) [2]. The generator network, $G(z)$, takes random noise and attempts to transform it into fake data that is indistinguishable from the original input data. The discriminator network, $D(z)$, attempts to discriminate between the generated data and the original data. Over time, both networks are gradually learning and improving.

The discriminator outputs a `logit` prediction, $\{0, 1\}$ for a given `X` that the generated data is either real or fake. Let D($\mathbf{x}$) represent the probability that $\mathbf{x}$ came from the data distribution p_{data}, rather than from p_g, the distribution of synthetic samples from the generator. The GAN trains the discriminator to maximize the probability of assigning the correct prediction to both synthetic and the original data. The generator is trained to transform random noise, $\mathbf{z}$, into synthetic data, represented by the function $\hat{\mathbf{x}} = G(\mathbf{z})$ where $\hat{\mathbf{x}}$ are the new generated synthetic data. Thus, the generator minimizes $log(1 - D(G(\mathbf{z}))$.

Now, we can define the objective function:

$$\begin{aligned} \min_G \max_D \mathbb{E}_{x \sim p_{data}}[logD(x)] + \mathbb{E}_{\hat{x} \sim p_g}[log(1 - D(\hat{x})] \\ = \mathbb{E}_x[logD(x)] + \mathbb{E}_z[log(1 - D(G(z))] \end{aligned}$$

After sufficient iterations of this minimax game, the distribution of synthetic samples p_g from the generator should be adequately close to the distribution of the true data, p_{data}.

Though GANs and Boltzmann machines are both generative models based on deep learning and neural networks, they have very different mechanisms. A Boltzmann machine is a symmetrically

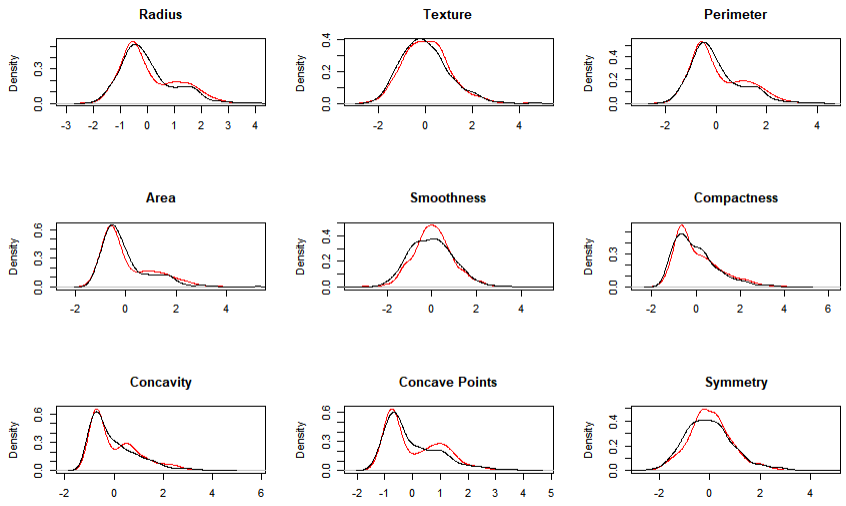

Figure 2: Empirical densities of synthetic data (red) compared to original data (black)

connected (undirected) network, allowing for both forward and backward connections between binary hidden and visible units. The configuration of the network defines its energy function. A GAN is more of an architecture, consisting of two neural networks whose purposes directly oppose each other. Furthermore, the generator is a directed network and so only allows for forward (not backward) connections when producing synthetic samples. The same applies to the discriminator. Thus, we can see that GANs and Boltzmann machines are very different.

3.2 Synthetic vs. Original Data

In applying the GAN to the breast cancer data set, we first standardized the data, then we implemented the algorithm in Tensorflow. We chose a Gaussian prior, $N(0, 6)$, as the random noise for the generator to learn the distribution of the original data, following the observed normality of the breast cancer data. The generator was constructed with two hidden layers (stages between the input and output which take the weighted input and transform it through an activation function to produce a viable output), each with 16 nodes (nodes simulate the behavior of neurons by connecting directly to an input variable and contributes to the output variable). The discriminator was constructed with three hidden layers, each with 16 nodes.

After 10,000 iterations, the synthetic and original data were difficult to differentiate from each other. For example, the empirical densities of the synthetic and original data were similar, with few exceptions (Figure 2). The densities of synthetic data occasionally showed higher peaks, such as in the case of smoothness and compactness. The synthetic data also introduced additional local maxima, as in the case of concavity.

It is important to note that the GAN is a random process; further simulations will produce different (albeit similar) results. However, after running the GAN with the same hyperparameters multiple times, we obtained results that were consistent with our original simulation.

4 Comparison of Graph Models

In order to evaluate the synthetic data's performance in maintaining the structure and properties of the original data, we ran stepwise forward model selection and a Bayesian method, birth-death Markov Chain Monte Carlo for Gaussian graphical models.

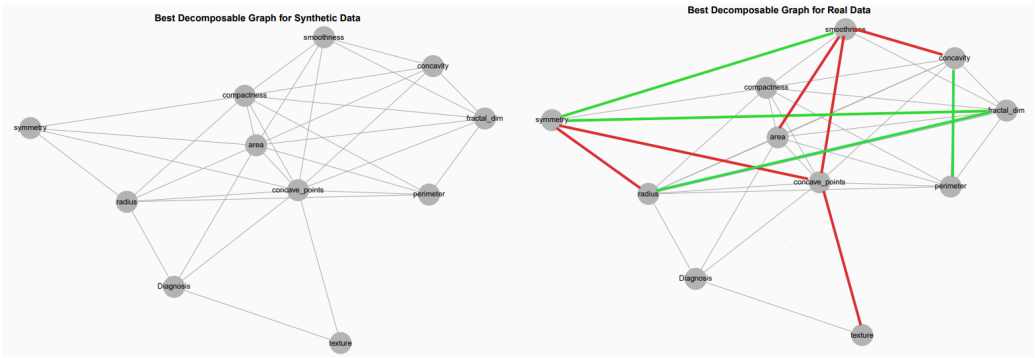

Figure 3: Best decomposable graphs for synthetic and real data, addition of edges in green and deletion of edges in red

4.1 Decomposable Graphs

Using the method from lecture, we ran stepwise forward model selection on both the synthetic and original data sets. As shown in Figure 3, the synthetic data added six edges that were not present in the model of the original data and lost four edges that were present in the original data. With how few variables are present in the dataset, the synthetic data performed rather poorly in producing a comparable model to the original data.

4.2 Bayesian Structural Learning

After running stepwise forward model selection, we aimed to validate the results by applying a more complex model to both datasets. For this part, we chose to implement birth-death Markov Chain Monte Carlo (BDMCMC) [5]. A traditional MCMC method samples from a desired posterior density by constructing a Markov chain that has that density as its stationary distribution. After sufficient iterations, the MCMC converges to the desired posterior.

The birth-death MCMC approach, based on a continuous time Markov process, is an alternative to traditional MCMC. Transitions to a larger dimension (adding edges) is a birth process, while transitions to a smaller dimension (deleting edges) is a death process. Furthermore, the time between jumps to a higher or lower dimension is modeled by a Poisson random variable. The relative rates at which births and deaths occur determines the stationary distribution of the process.

At base, a zero-mean Gaussian graphical model is defined with respect to graph G:

$$\mathcal{M}_{\mathrm{G}} = \left\{ N_{\mathrm{p}}(0, \Omega^{-1}) \mid \Omega \in \mathbb{P}_G \right\}$$

where $\mathbb{P}_G$ is the set of all valid positive-definite precision matrices. Furthermore, a G-Wishart conjugate prior is placed on precision matrix Ω, with density

$$p(\Omega \mid G) = I_G(b, D)^{-1} |\Omega|^{(b-2)/2} exp\left\{ -\frac{1}{2} tr(D\Omega) \right\} 1_{\{\Omega \in \mathbb{P}_G\}}$$

where I_G is the normalizing constant, and b and D are parameters of the density. Finally, a truncated Poisson prior is placed on the graph itself:

$$p_\gamma(G) \propto \frac{\gamma^{|E|}}{|E|!} \quad \forall G = (V, E) \in \mathcal{G}$$

where $|E|$ is the size of the graph. The BDMCMC algorithm is as follows. For S iterations, given graph $G = (V, E)$, calculate the birth and death rates separately, determine the waiting time conditional on these rates, simulate the type of jump (higher or lower dimension), and finally conditional on the type of jump, sample from the new precision matrix.

After applying the BDMCMC method to both our generated synthetic data and the original data, the differing results became apparent (Figure 4). The graph produced by the synthetic data is much more dense (containing 27% of all possible edges) than the graph produced by the original data (containing

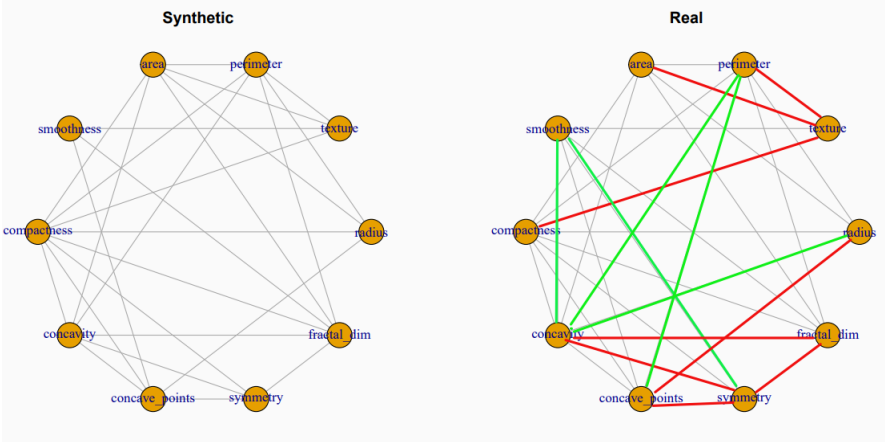

Figure 4: Graph structure learning using birth-death MCMC with both synthetic and real data, addition of edges in green and deletion of edges in red

only 20% of all possible edges). The synthetic data introduced 8 additional edges, while deleting only 5. This might suggest that the GAN introduced new hidden correlations between certain variables, but the near-identical similarity between the correlation matrices of the synthetic and original data suggests otherwise. All in all, since the synthetic data produced a very different graph than the original data, suggesting that generated data from a GAN may not be suitable for training some types of models.

5 Conclusions and Future Work

Future work could involve comparing other methods such as graphical lasso, however given the poor performance of GANs in maintaining the model structure as the original data, we do not expect to see more favorable results from graphical lasso. In evaluating GANs as a potential way of privatizing data, we left checking its success in privatizing the data to future work, however since the synthetic data does not result in a similar model as the real data, this future work is unnecessary. However, the quality of the synthetic data produced may be improved, either by tuning various hyperparameters in the GAN, or using an alternative method to make synthetic data. Training a GAN is notoriously difficult, and work adjusting the parameters might improve the performance of GANs in replicating the model structure of the original data.

References

[1] Chawla, N., Bowyer, K., Hall, L. and Philip Kegelmeyer, W. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *J. Artif. Intell. Res. (JAIR).* **16**. 321-357.

[2] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio., Y. (2014). Generative Adversarial Nets. In *NIPS'2014*.

[3] Li, H., Xiong, L., Zhang, L., Jiang, X. (2014). DPSynthesizer: Differentially private data synthesizer for privacy preserving data sharing. In Proc. of the VLDB Endowment, 7(13). 1677-1680.

[4] McGuiness, K. *Deep Learning for Computer Vision: Generative models and adversarial training* (UPC 2016).

[5] Mohammadi, A., and Wit, E. C. (2015). Bayesian structure learning in sparse Gaussian graphical models. *Bayesian Analysis*, **10**(1), 109-138.

[6] Surendra, H., and Mohan, H.S. (2017). A Review of Synthetic Data Generation Methods for Privacy Preserving Data Publishing. *International Journal of Scientific & Technology Research*, **6**(3), 95-101.

[7] Woolberg, W.H., Street, W.N., and Mangasarian, O.L. (1995). Breast cancer Wisconsin (diagnostic) data set. UCI Machine Learning Repository.