Population Synthesis with Deep Generative Models - is it worth it? Exploring new models and metrics

Vianey Darsel*[1], Etienne Côme[1], and Latifa Oukhellou[1]

[1]COSYS-GRETTIA, Gustave Eiffel University, France

Short summary

Population synthesis is a fundamental element for activity-based simulation models, consisting in generating a full synthetic population from a restricted dataset. Even if many methods have been studied, diffusion models — the current state-of-the-art Deep Generative Models (DGM) for tabular data — have not yet been thoroughly investigated. This paper addresses this gap and conducts a benchmark comparison with previous models, including investigations on the impact of encoding choices. To build a robust benchmark, metrics to assess the distribution, originality and realism of a generated population are redefined. Furthermore, this benchmark is conducted using a large French census dataset allowing an accurate evaluation. By considering two scenarios of training set size —representing typical use cases — this study offers generalizable guidelines. Notably, with appropriate data encoding, diffusion models outperform other DGMs. However, Bayesian Networks, a probabilistic model, demonstrate comparable or superior performances while requiring fewer computational resources, making them attractive in similar applications.

Keywords: Activity-based modeling, Deep Generative Models, Diffusion Models, Methodology, Population Synthesis.

1 Introduction & Literature

Activity-based models are highly effective tools for transport simulation, capable of modeling the movements of an entire population within a specific area at an individual level (Horni et al., 2016). These models account for individual decision-making and travel behaviour. To operate, they require a synthetic population as input (Hörl & Balac, 2021).

The methods used for population synthesis have evolved. Early approaches, known as reproduction models, relied on copying individuals from training data to reconstruct the total population (Beckman et al., 1996; Voas & Williamson, 2000; Ye et al., 2009). However, these methods are limited by their inability to generate new, out-of-sample individuals, making them less effective. Subsequently, probabilistic models were introduced, learning the structure and probabilistic relationships between attributes (Farooq et al., 2013; Sun & Erath, 2015). More recently, advancements in Deep Learning have led to the emergence of Deep Generative Models (DGMs), which have demonstrated state-of-the-art performance in generating tabular data (Xu et al., 2019; Armanious et al., 2020; Zhang et al., 2023). These models have also been successfully applied to population synthesis (Borysov et al., 2019; Garrido et al., 2020; Kim & Bansal, 2022). Despite their success, diffusion models, which represent the current state-of-the-art generative modeling, have not been thoroughly explored for population synthesis. To address this gap, we propose a rigorous diffusion-based model tailored for population synthesis. Additionally, we perform a comprehensive comparison of DGMs and probabilistic models, building on previous work that has only partially addressed this comparison (Borysov et al., 2019).

Evaluating a synthetic population is also critical. In reproduction and probabilistic models, only the **distribution** of the generated population was compared to the one of the true population. Recent contributions have also introduced new challenges in assessing the quality of synthetic populations, particularly through the concepts of *sampling zeros* and *structural zeros* (Borysov et al., 2019). *Sampling zeros* pertain to the model's ability to generate data points that, while absent from the training dataset, are present within the testing dataset. This capability is linked to the synthetic population's **originality**. On the other hand, *structural zeros* refer to instances that should not be generated due to inherent incompatibilities, encapsulating the criterion of **realism**. Consequently, three key criteria emerge for evaluating synthetic populations: distribution, originality, and realism.

Defining the metrics to evaluate the criteria is a critical challenge. To address this, this work examines existing evaluation metrics and redefines certain ones to better align with the three introduced criteria.

When addressing population synthesis, encoding attributes is another important factor to consider. Indeed, a synthetic individual is composed of categorical and continuous attributes. In reproduction and probabilistic models, the choice has always been to create bins and discretize continuous attributes to deal only with categorical attributes (Beckman et al., 1996; Farooq et al., 2013; Sun & Erath, 2015). With DGMs, considering continuous attributes (Borysov et al., 2019), and a continuous embedding for categorical attributes (Kim & Bansal, 2022) have been explored. This paper further examines the optimal encoding of each attribute.

The literature review highlights several gaps, which form the basis for the contributions of this paper, outlined as follows:

- Evaluate diffusion models for population synthesis
- Present the first extensive benchmark with both DGMs and Probabilistic models
- Study the impact of different data encodings in the context of DGMs
- Define a new metric that quantifies better *structural zeros*

To investigate the different points, we have access to the French census data, which comprises around a third of the total population. Thus, thanks to its size, this dataset is a good case study to evaluate a synthetic population.

Overall, this paper aims to highlight good practices for population synthesis in terms of models, data encoding for DGMs, and evaluation metrics.

2 METHODOLOGY

We aim to compare synthetic populations generated from different models. To that extent, we first sample a population with different DGMs and data encodings using different training data sizes. Then, for each training data size, we compare the performance of the best DGM with probabilistic models. All the comparisons are done on quantitative metrics.

Models

For the probabilistic models, we consider the two algorithms that have been studied in population synthesis, Markov Chain Monte Carlo (Farooq et al., 2013; Saadi et al., 2016) and Bayesian Networks (Sun & Erath, 2015; Choi et al., 2017; Sun et al., 2018). For DGMs, Variational Auto-Encoders (Borysov et al., 2019; Garrido et al., 2020; Kim & Bansal, 2022), Generative Adversarial Networks (Garrido et al., 2020; Kim & Bansal, 2022), and a diffusion model (Kang et al., 2023) have been studied for population synthesis. As diffusion models have never been appropriately presented in a population synthesis context, we give a larger description of this model. Some DGMs require an embedding network, which will be detailed later in the paper.

Markov Chain Monte Carlo (MCMC) Gibbs Sampling, a MCMC method, has proven to be highly effective for sampling a distribution (Geman & Geman, 1984; Gelfand & Smith, 1990). One attribute is updated at the time conditionally to all other attributes. We investigate two ways of estimating the conditional distributions: the observed frequency and the Bayesian Posterior.

Bayesian Networks (BN) BNs consist in constructing a Probabilistic Graphical Model over the attributes (Heckerman, 1998). Once the structure is learned, the sampling is done with the observed conditional frequencies. We investigate two methods for building the graphical model: a tree structure using the Chow-Liu algorithm (Chow & Liu, 1968), and hill-climbing with the BIC (Koller & Friedman, 2009), that allows non-tree structure.

Variational AutoEncoder (VAE) VAE is a DGM where data is sampled in a low dimension, and then upscaled to the original dimension (Kingma, 2013). The idea is that the underlying data dependencies allow modeling in low dimensions. We consider two versions of VAE, with and without the Transformer embedding.

Generative Adversarial Network (GAN) A GAN operates as a two-player minimax game between a generator and a discriminator (Goodfellow et al., 2014). The generator aims to produce realistic samples, whereas the discriminator aims to estimate whether the input data was generated by the generator or comes from the training set. We include two versions of GAN in the benchmark, with and without the Transformer embedding.

Diffusion Adding noise to a vector is a straightforward process. By using an original vector and its noisy counterpart, a model can be trained to estimate the noise in any given vector. This principle underlies diffusion models, which begin with a noisy vector and iteratively reduce the noise until achieving a noise-free vector. In Figure 1, we show on a toy example how the noising and denoising are progressively done, with a two-dimensional variable and three-time steps. The sampling trajectory of a generated sample is shown in purple.

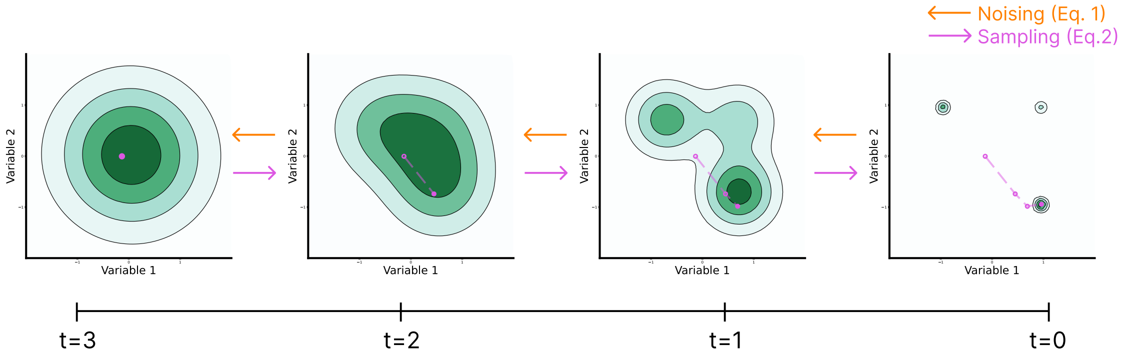


Figure 1: Toy example of sampling a data point with two attributes. The green area represents the probability law at each time step, whereas the purple point shows the sampling trajectory from a data point.

In the previous application of diffusion models to population synthesis (Kang et al., 2023), the architecture used was designed for images. Images are structured data, whereas a synthetic population should be considered as a tabular data problem. Thus, we use solutions designed for tabular data by using a continuous embedding, with a Transformer network, and linear layers to sample mixed-type data (Zhang et al., 2023).

For formal explanations on diffusion models, we introduce the notation $\mathbf{X}^t$ and p_t corresponding to the vector $\mathbf{X}$ and its distribution p, at the time variable $t \in [0, T]$. At $t = 0$, we have the original data $\mathbf{X} = \mathbf{X}^0$ and distribution $p = p_0$, and at $t = T$, we have the noise data.

The two phases of the diffusion model, noising and sampling, have been modeled into equations by Song et al. (2021). Thus, the noising process is given by:

$$d\mathbf{X}^t = \mathbf{f}(\mathbf{X}^t, t)dt + g(t)d\mathbf{w} \tag{1}$$

where:

- $\mathbf{w}$ is a standard Wiener process.
- $\mathbf{f}$ is the drift function.
- g is the diffusion coefficient.

The denoising process is written as follows:

$$d\mathbf{X}^t = [\mathbf{f}(\mathbf{X}^t, t) - g(t)^2 \nabla_{\mathbf{X}} \log p_t(\mathbf{X}^t)]dt + g(t)d\bar{\mathbf{w}} \tag{2}$$

where:

- $\bar{\mathbf{w}}$ is a standard Wiener process.

The functions $\mathbf{f}$ and g are selected by the experimenter. In Equation 2, the main challenge lies in estimating the score function $s_\theta(\mathbf{X}^t, t) := \nabla_{\mathbf{X}^t} \log p_t(\mathbf{X}^t)$, what is done with a neural network.

Equations 1 and 2 enable transitions backward or forward in time while maintaining the distribution of p_t.

In this paper, we use an adaptation from Song et al.'s model by Karras et al. (2022), which improves the design with faster sampling and global training.

Data encoding

For probabilistic models, we treat all attributes as categorical since these models are specifically designed to handle this type of data. In contrast, for DGMs, the computations within the models are performed using continuous values. To handle the conversion from categories to continuous representations, we propose to use an embedding network based on a Transformer (Devlin et al., 2019). Moreover, we are interested in the impact of considering variables as continuous or categorical. Thus, we compare three encodings for numeric attributes:

- `All continuous` considering all numeric attributes as continuous.
- `Only age continuous` considering only the age attribute as continuous and the other attributes as categorical. The motivation is that age has the highest possible modalities (see Table 1).
- `All categorical` considering all numeric attributes as categorical, as for probabilistic models.

Evaluation metrics

A generated synthetic population is evaluated by comparing it with a testing dataset. From the analysis of our literature, we propose to evaluate a synthetic population according to 3 criteria:

- **Faithful distribution**: This criterion evaluates whether the model respects the multivariate distributions from the original data. We use the Standardized Rooted Mean Squared Error (SRMSE) on the trivariate distribution to evaluate this criterion.

  For each triplet of feature (i, j, k) among $\binom{\{1,\ldots,n\}}{3}$ — the $\binom{n}{3}$ possible triplets—, the mean square error between the frequencies of each possible modalities of triplets values $x^{ijk} \in \mathcal{X}^{ijk}$ in the generated and testing dataset are computed, and then the average of the scores is computed. As a result, we have the following metric:

  $$SRMSE_3(X_{gen},X_{test})=\frac{1}{\binom{n}{3}}\sum_{(i,j,k)\in\binom{\{1,\ldots,n\}}{3}}\sqrt{\sum_{x^{ijk}\in\mathcal{X}^{ijk}}(f_{gen}(x^{ijk})-f_{test}(x^{ijk}))^2|\mathcal{X}^{ijk}|} \tag{3}$$

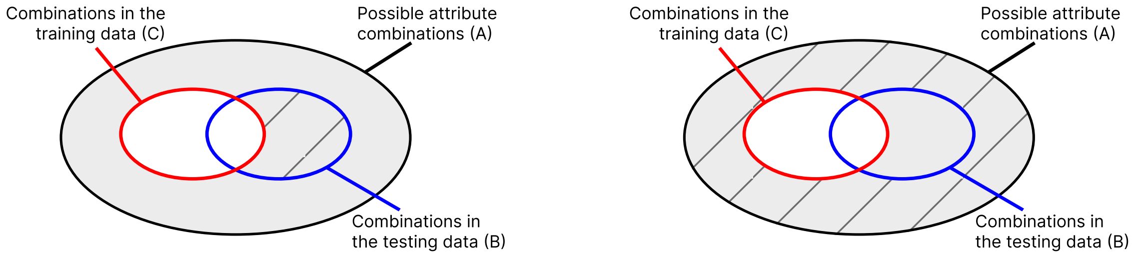
  where f_{set} corresponds to the observed frequency in the corresponding set.

- **Originality**: This criterion evaluates the generative capacities of the model. It corresponds to the *sampling zeros* problem. The metric we choose here is the share of generated samples that are not in the training data but in the testing data. The *sampling zeros* rate is the share of generated samples that belongs to the hashed part of Figure 2a among samples in the grey area. To ease the interpretation of the metric and align it with the other criteria, we adjust it to follow a minimizing objective. Thus, using the notation from Figure 2, we can derive the metric Share of Samples Out of Training and Testing (SSOTT) as follows:

  $$SSOTT(X_{gen}, X_{train}, X_{test}) = 1 - \frac{\sum_{x\in X_{gen}} \mathbb{1}_{x\in B\cap\overline{C}}}{\sum_{x\in X_{gen}} \mathbb{1}_{x\in\overline{C}}} = \frac{\sum_{x\in X_{gen}} \mathbb{1}_{x\in\overline{B\cup C}}}{\sum_{x\in X_{gen}} \mathbb{1}_{x\in\overline{C}}} \tag{4}$$

  SSOTT computes the share of generated samples out of the training data also out of testing data, which corresponds to the share of samples in the hashed area among samples in the grey areas from Figure 2b.

- **Realism**: Our goal is to ensure the generated population is realistic, avoiding the presence of *structural zeros*. However, the existing definition (and its resulting metric) proposed in the literature by Borysov et al. (2019) poses challenges as the number of features increases. Indeed, a generated sample is labelled as a *structural zero* if its entire combination of attributes is absent from both the training and testing sets. As the number of features grows, the number of possible attribute combinations increases drastically, requiring an impractically large amount of test data to obtain a robust estimate of realistic samples. Therefore, we propose a new definition for *structural zeros*. We consider that a generated individual is labeled as a *structural zero* if a couple of instances are unseen in the data, e.g., a child with a driver's license. To evaluate the realism, we measure the Share of Samples with a Couple of Instances that is Out of Testing data (SSCIOT). Such a definition does not suffer from the curse of dimensionality, and we hypothesize that this is sufficient to detect unrealistic combinations.



(a) Intuitive definition of the Originality metric

(b) Redefinition of the intuitive metric to get a minimizing objective

Figure 2: Venn diagram explaining the definition of the originality metric

Name	Type	Number of instances	min	max
Age	integer	120	0	120
Sex	binary	2		
Last diploma	category	5		
Number of persons in the household	integer	25	0	64
Type of household	category	5		
Type of professional activity	category	9		
Family link	category	5		
Married	boolean	2		
Department	category	8		
Number of cars	integer	4	0	3
Socio-professional category	category	8		
Type of accommodation	category	7		

Table 1: Description of the attributes used for the study. The minimum and maximum values are given only for numerical attributes.

3 RESULTS & DISCUSSION

Data

Population syntheses are typically created using census data or Household Travel Surveys (HTS). Census data provides socioeconomic profiles of individuals, while HTS adds insights into travel preferences and activity chains, making them more suitable for a synthetic population in a transport framework. A key distinction between these data sources lies in the dataset sizes. While the exact size varies by country, census data generally represents between 1% and 5% of the population, compared to around 0.03% for HTS.

For better generalization, we train our models on two different scenarios, with 0.03% and 1% of the total population, respectively, to replicate the settings of HTS and typical census data.

For our study, we use the French census data from 2015 on the Île-de-France area. We restrict ourselves to the generation of the relevant socioeconomic attributes from census data. We limit our study to twelve attributes based on their relevance and on the literature (Farooq et al., 2013; Hörl & Balac, 2021) — three numerical and nine categorical attributes. The attributes are described in Table 1.

A population is generated with each trained model and then evaluated by comparison with the true population, which is approximated by a test population composed of 23% of the total population. As the testing set is large, it is a good approximation of the total population.

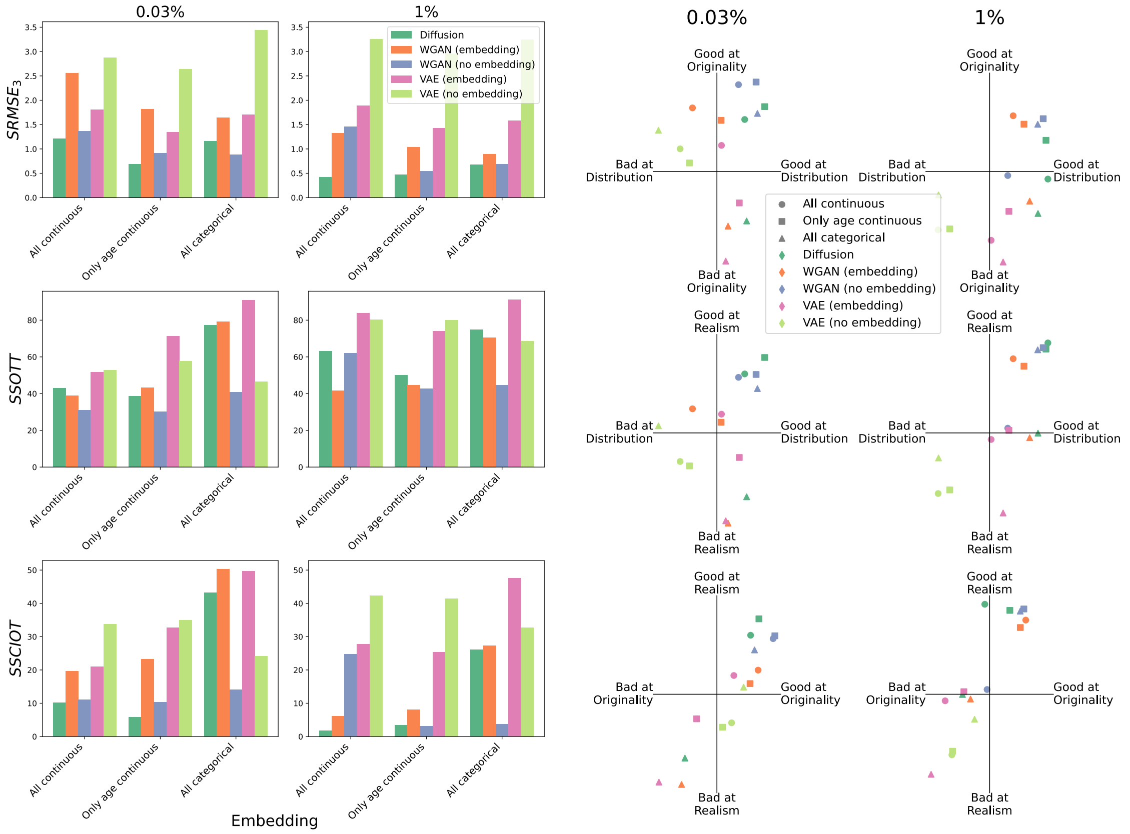
Experiments

We first compare DGMs and their encodings with each other for the two scenarios. From that, we determine the best model and encoding for each scenario. In this study, we did not interest us in the models' tuning, we use architecture similar to previous contributions (Borysov et al., 2019; Kim & Bansal, 2022; Zhang et al., 2023). Thus, the number of parameters of DGMs varies from

100,000 (for VAE) to 10,000,000 (for diffusion). Then, it is interesting to compare the performance of DGMs with classical models that require fewer resources and were the previous state-of-the-art models in population synthesis. Thus, we compare the best DGM with probabilistic models, for which the question of encoding does not arise.

Comparison of DGMs

The different measures are given in Figure 3, and Table 2. In Figure 3a and 3b, the left column (resp. the right column) indicates the results for a training population of 0.03% (resp. 1%).



(a) Bar plots for each metric. The smaller the bar is, the better the model performs.

(b) Cross performance between two criteria

Figure 3: Comparison of DGMs and encodings on the different evaluation criteria and at different training sizes.

We observe that diffusion models perform better than the other models for distribution and realism, except in some cases for the `All categorical` encoding. WGAN without embedding tends to perform better for originality, although the result of the best diffusion model in originality is similar. In contrast, VAE almost always performs the worst on the different metrics, and WGAN with embedding is generally worse than the no-embedding version. We see that the encoding of the data has an important impact on almost all models.

At 0.03%, using a diffusion model with an `Only age continuous` embedding is a good choice, as it has the best overall performances in realism and originality, and the relative gap for the last criterion with the best configuration (21.4%) is acceptable. This choice is strengthened by Figure 3b. In this figure, the point of a model should be as top right as possible. At 0.03%, diffusion with `Only age continuous` stands out on each plot.

At 1%, the results are not as clear as at 0.03% in Figure 3b. However, diffusion model with an `Only age continuous` embedding is still the best trade-off as diffusion with `All continuous` or `Only age continuous` embeddings are the best models in terms of distribution and realism. Still, the `Only age continuous` has significantly better results for originality, comparable to the best models.

Finally, we observe that diffusion models outperform other models. More precisely, diffusion model with `Only age continuous` embedding is the best model for both training data sizes.

Training size of 0.03% of the total population

DGM	Continuous data representation	Distribution $SRMSE$	Originality $SSOTT$	Realism $SSCIOT$
Diffusion	`All continuous`	1.21	42.91%	10.16%
	`Only age continuous`	**0.693**	38.56%	**5.74%**
	`All categorical`	1.16	77.3%	43.22%
WGAN (embedding)	`All continuous`	2.56	38.98%	19.55%
	`Only age continuous`	1.82	43.2%	23.17%
	`All categorical`	1.64	79.05%	50.31%
WGAN (no embedding)	`All continuous`	1.37	31.11%	11.08%
	`Only age continuous`	0.915	**30.22%**	10.31%
	`All categorical`	0.881	40.86%	14.11%
VAE (embedding)	`All continuous`	1.81	51.69%	20.99%
	`Only age continuous`	1.35	71.18%	32.64%
	`All categorical`	1.7	90.9%	49.68%
VAE (no embedding)	`All continuous`	2.88	52.86%	33.72%
	`Only age continuous`	2.64	57.63%	34.91%
	`All categorical`	3.44	46.57%	24.12%

Training size of 1% of the total population

DGM	Continuous data representation	Distribution $SRMSE$	Originality $SSOTT$	Realism $SSCIOT$
Diffusion	`All continuous`	**0.422**	63.09%	**1.82%**
	`Only age continuous`	0.471	49.97%	3.47%
	`All categorical`	0.678	74.76%	26.1%
WGAN (embedding)	`All continuous`	1.32	**41.64%**	6.08%
	`Only age continuous`	1.04	44.5%	8.08%
	`All categorical`	0.893	70.57%	27.32%
WGAN (no embedding)	`All continuous`	1.46	61.9%	24.76%
	`Only age continuous`	0.543	42.64%	3.09%
	`All categorical`	0.685	44.54%	3.68%
VAE (embedding)	`All continuous`	1.89	83.84%	27.8%
	`Only age continuous`	1.43	74.04%	25.35%
	`All categorical`	1.58	91.2%	47.58%
VAE (no embedding)	`All continuous`	3.26	80.31%	42.34%
	`Only age continuous`	2.96	79.98%	41.37%
	`All categorical`	3.25	68.45%	32.76%

Table 2: Comparison of the different DGMs and encodings on three criteria. For each metric, the optimal value is the smallest one and is highlighted in bold.

Comparison with Probabilistic Models

The results are presented in Table 3. We observe that the performance of MCMC models is significantly lower. As the frequency approach can only reproduce samples from the training set, the realism of the generated population is optimal, but its distribution is the worst by far. Regarding the Bayesian approach, it acts as a uniform random variable when the training set is too small, all combinations are possible, in that case, making $SSCIOT$ exploding. Therefore, we concentrate on the comparison between the BN and diffusion model.

Training size of 0.03% of the total population

Model	Distribution $SRMSE$	Originality $SSOTT$	Realism $SSCIOT$
MCMC (freq.)	28.9	NC	**0%**
MCMC (Bayesian)	2.81	99.99%	96.45%
BN (hill)	0.773	50.61%	4.62%
BN (tree)	0.79	48.67%	0.69%
Diffusion	**0.693**	**38.56%**	5.74%

Training size of 1% of the total population

Model	Distribution $SRMSE$	Originality $SSOTT$	Realism $SSCIOT$
MCMC (freq.)	6.34	NC	**0%**
MCMC (Bayesian)	2.84	100.0%	98.07%
BN (hill)	**0.432**	**46.29%**	0.37%
BN (tree)	0.676	60.54%	3.4%
Diffusion	0.471	49.97%	3.47%

Table 3: Comparison of the best DGM with probabilistic models on three criteria. For each metric, the optimal value is the smallest one and is highlighted in bold.

Model	Training + Sampling Time
Diffusion	78 minutes
BN (greedy)	9 seconds
BN (tree)	7 seconds

Table 4: Time for training and sampling for the different models for a training set of 1%. For diffusion, 76 minutes are spent for the training.

At 0.03%, diffusion is slightly better than BNs for distribution and originality but performs much worse than BN with tree structure when regarding realism. At 1%, BN with a greedy search of structure is the best model for all criteria. The difference is small for distribution and originality with diffusion but significant for realism. We note that, for each training size, one BN almost gets a perfect score for the realism, with only 0.69% of samples with incompatible attributes at 0.03% for the tree structure and 0.37% of samples with incompatible attributes at 1% for the greedy search.

It seems that the best DGM does not significantly perform better than BNs. Thus, BNs seem to be the most promising network for synthetic population. To strengthen this result, the training and sampling time are given in Table 4. We see that training a diffusion model is time-consuming. Regarding the slight benefit at 0.03%, it is not worth the effort of training such a big model in our cases of Population Synthesis.

4 Conclusions & Perspectives

We have provided three criteria to evaluate a synthetic population and adapted existing metrics to better suit the defined criteria. Taking advantage of the large census data available in France, we compare different Deep Generative Models (DGMs) with probabilistic models. All in all, for experimental framework similar to our experiments, Bayesian Networks (BNs) are recommended to generate a synthetic population. BNs perform similarly to the best combination of DGM with a data encoding, a diffusion model with `Only age continuous` encoding. However, the resources and the time required to train and sample with DGM do not justify the small gap in performances with BN.

We restrict our study to 12 attributes, because no other attribute is relevant in the available data. Increasing the number of attributes increases the difficulty of the problem. It will be interesting to see the evolution of these results on a dataset with more attributes, especially since VAE is better than BN with many attributes (Borysov et al., 2019). DGMs could be more interesting in more complex tasks, in the generation pipeline, for instance, to generate an activity chain or to allocate geographical positions to the individuals. Moreover, this study is not interested in the impact of the size of the training data.We note that in our results, the originality is not improved by increasing the training size, unlike other criteria. This is due to the definition of the metric: the set C from Figure 2 gets bigger when the dataset gets bigger. The hyperparameters of DGMs are not discussed in this work, and a rigorous hyperparameter research methodology is necessary for applicative use cases. All these comments are interesting roots for future work.

References

Armanious, K., Jiang, C., Fischer, M., Küstner, T., Hepp, T., Nikolaou, K., ... Yang, B. (2020). Medgan: Medical image translation using gans. *Computerized medical imaging and graphics*, *79*, 101684.

Beckman, R. J., Baggerly, K. A., & McKay, M. D. (1996, November). Creating synthetic baseline populations. *Transportation Research Part A: Policy and Practice*, *30*(6), 415–429. Retrieved 2024-04-05, from `https://www.sciencedirect.com/science/article/pii/0965856496000043` doi: 10.1016/0965-8564(96)00004-3

Borysov, S. S., Rich, J., & Pereira, F. C. (2019, September). How to generate micro-agents? A deep generative modeling approach to population synthesis. *Transportation Research Part C: Emerging Technologies*, *106*, 73–97. Retrieved 2024-04-05, from `https://www.sciencedirect.com/science/article/pii/S0968090X1831180X` doi: 10.1016/j.trc.2019.07.006

Choi, E., Biswal, S., Malin, B., Duke, J., Stewart, W. F., & Sun, J. (2017, November). Generating Multi-label Discrete Patient Records using Generative Adversarial Networks. In *Proceedings of the 2nd Machine Learning for Healthcare Conference* (pp. 286–305). PMLR. Retrieved 2024-05-13, from `https://proceedings.mlr.press/v68/choi17a.html` (ISSN: 2640-3498)

Chow, C., & Liu, C. (1968). Approximating discrete probability distributions with dependence trees. *IEEE transactions on Information Theory*, *14*(3), 462–467.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019, May). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.* arXiv. Retrieved 2024-04-18, from `http://arxiv.org/abs/1810.04805` (arXiv:1810.04805 [cs]) doi: 10.48550/arXiv.1810.04805

Farooq, B., Bierlaire, M., Hurtubia, R., & Flötteröd, G. (2013, December). Simulation based population synthesis. *Transportation Research Part B: Methodological*, *58*, 243–263. Retrieved 2024-04-05, from `https://www.sciencedirect.com/science/article/pii/S0191261513001720` doi: 10.1016/j.trb.2013.09.012

Garrido, S., Borysov, S. S., Pereira, F. C., & Rich, J. (2020, November). Prediction of rare feature combinations in population synthesis: Application of deep generative modelling. *Transportation Research Part C: Emerging Technologies*, *120*, 102787. Retrieved 2024-04-05, from `https://www.sciencedirect.com/science/article/pii/S0968090X20306975` doi: 10.1016/j.trc.2020.102787

Gelfand, A. E., & Smith, A. F. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American statistical association*, *85*(410), 398–409.

Geman, S., & Geman, D. (1984). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence*(6), 721–741.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, *27*.

Heckerman, D. (1998). A tutorial on learning with bayesian networks. *Learning in graphical models*, 301–354.

Horni, A., Nagel, K., & Axhausen, K. W. (2016). Introducing matsim. In *The multi-agent transport simulation matsim* (pp. 3–7). Ubiquity Press.

Hörl, S., & Balac, M. (2021, September). Synthetic population and travel demand for Paris and Île-de-France based on open and publicly available data. *Transportation Research Part C: Emerging Technologies*, *130*, 103291. Retrieved 2024-04-05, from `https://www.sciencedirect.com/science/article/pii/S0968090X21003016` doi: 10.1016/j.trc.2021.103291

Kang, J., Kim, Y., Imran, M. M., Jung, G.-S., & Kim, Y. B. (2023). Generating population synthesis using a diffusion model. In *2023 winter simulation conference (wsc)* (pp. 2944–2955).

Karras, T., Aittala, M., Aila, T., & Laine, S. (2022, October). *Elucidating the Design Space of Diffusion-Based Generative Models.* arXiv. Retrieved 2024-06-13, from `http://arxiv.org/abs/2206.00364` (arXiv:2206.00364 [cs, stat]) doi: 10.48550/arXiv.2206.00364

Kim, E.-J., & Bansal, P. (2022, August). *A Deep Generative Model for Feasible and Diverse Population Synthesis.* arXiv. Retrieved 2024-04-05, from `http://arxiv.org/abs/2208.01403` (arXiv:2208.01403 [cs, stat]) doi: 10.48550/arXiv.2208.01403

Kingma, D. P. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Koller, D., & Friedman, N. (2009). *Probabilistic graphical models: principles and techniques.* MIT press.

Saadi, I., Mustafa, A., Teller, J., Farooq, B., & Cools, M. (2016, August). Hidden Markov Model-based population synthesis. *Transportation Research Part B: Methodological*, *90*, 1–21. Retrieved 2024-04-05, from `https://www.sciencedirect.com/science/article/pii/S0191261515300904` doi: 10.1016/j.trb.2016.04.007

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., & Poole, B. (2021). Score-based generative modeling through stochastic differential equations. In *International conference on learning representations.* Retrieved from `https://openreview.net/forum?id=PxTIG12RRHS`

Sun, L., & Erath, A. (2015, December). A Bayesian network approach for population synthesis. *Transportation Research Part C: Emerging Technologies*, *61*, 49–62. Retrieved 2024-04-05, from `https://www.sciencedirect.com/science/article/pii/S0968090X15003599` doi: 10.1016/j.trc.2015.10.010

Sun, L., Erath, A., & Cai, M. (2018, August). A hierarchical mixture modeling framework for population synthesis. *Transportation Research Part B: Methodological*, *114*, 199–212. Retrieved 2024-04-05, from `https://www.sciencedirect.com/science/article/pii/S0191261517308615` doi: 10.1016/j.trb.2018.06.002

Voas, D., & Williamson, P. (2000). An evaluation of the combinatorial optimisation approach to the creation of synthetic microdata. *International Journal of Population Geography*, *6*(5), 349–366. doi: 10.1002/1099-1220(200009/10)6:5<349::AID-IJPG196>3.0.CO;2-5

Xu, L., Skoularidou, M., Cuesta-Infante, A., & Veeramachaneni, K. (2019). Modeling tabular data using conditional gan. *Advances in neural information processing systems*, *32*.

Ye, X., Konduri, K., Pendyala, R., Sana, B., & Waddell, P. (2009, January). Methodology to match distributions of both household and person attributes in generation of synthetic populations.

Zhang, H., Zhang, J., Shen, Z., Srinivasan, B., Qin, X., Faloutsos, C., … Karypis, G. (2023, October). Mixed-Type Tabular Data Synthesis with Score-based Diffusion in Latent Space.. Retrieved 2024-04-05, from `https://openreview.net/forum?id=4Ay23yeuz0`