Deep generative models for combined population and job synthesis

Sergio H. Garrido M., Stanislav S. Borysov, Francisco C. Pereira, Jeppe Rich
Department of Management, Technical University of Denmark,
DTU, Kgs. Lyngby, Denmark

Abstract

Population synthesis is an essential step when modelling transport demand. In this paper, we consider a problem of generating population on a very detailed level which includes origins and destinations of the agents among other socio-demographic characteristics. To model this high-dimensional distribution, we propose to use generative approaches based on artificial neural networks from the deep learning area. As a case study based on a large travel survey data from Denmark, we show that the Generative Adversarial Network (GAN) and the Variational AutoEncoder (VAE) are capable of producing synthetic populations which statistical properties are in good agreement with the observed data.

1 Introduction

Models for population synthesis aims at generating samples of individuals for application in agent-based transport demand models. From a model perspective, the importance of the population synthesis stage cannot be overstated. Due to the sequential structure of most transport models, errors in the population synthesis stage will propagate in all subsequent stages of a model system. This may include models for car ownership, transport demand and choice of route.

Historically, it has been common to consider population synthesis as a deterministic matrix fitting problem. This is accomplished by fitting a population matrix based on; i) a starting solution and ii) known future margins, typically based on iterative proportional fitting [Deming and Stephan, 1940]. As a final stage, agents are drawn at random from the population matrix in order to accommodate an agent-based representation in contrast to the prototypical agent representation of the population matrix. In the process of generating appropriate synthetic populations, several challenges have been acknowledged in the literature. One challenge relates to the matrix fitting stage and how this may be carried out for very sparse matrices. Another, and slightly related challenge, is related to the re-sampling stage from one or more jointly related population matrices, e.g. how to sample consistently from separate matrices for individuals and households. Yet another challenge, which is the topic of this paper,

is concerned with the problem of generating appropriate starting solutions for any population framework. In other words, we restrain ourselves from the forecasting problem and focus on the sole problem of generating a 'baseline' pool of individuals with the same statistical properties as those in an original sample. Solving this challenge is important for several reasons. First, if a very detailed representation of the population is required, it will require large samples to cover the distribution. Typically, this will lead to the occurrence of types of individuals that are missing in the sample, not because they have a probability of zero of being selected, but because they occasionally were not included in the sample scheme. This is a problem when trying to project a future population because the likelihood of certain socio-economic combinations may evolve from being unlikely to more likely over time. The problem of missing information in the context of population synthesis was considered early in [Beckman et al., 1996] who suggested a heuristic approach to the problem. Another limitation of relying solely on a 'ground truth' sample is that it becomes difficult to investigate robustness with respect to the ground truth. A more general approach will be to consider the sample as one realisation from a ground truth distribution. By building a generic probabilistic framework for the ground truth distribution, learned from the original sample, it becomes possible to generate a variety of different samples from which robustness can be investigated.

In this paper, we approach the population synthesis problem from a generic probabilistic framework perspective. Furthermore, we extent the population synthesis problem to include detailed socio-economic characteristics of individuals including the zone of residential location, a specific sector description of a person's job and its location. In other words, the synthesis will render an indirect "commuter matrix", that is, the matrix of origin and destination of people when travelling from home to work. The paper extends existing literature in a number of ways. The paper is the first paper to consider the joint formation of residential location, jobs location and its geographical distribution across sectors. As these dimensions cannot be enumerated on an ordinal scale, it introduces non-trivial scalability challenges. Moreover, the paper is pioneering in the application of the Generative Adversarial Network (GAN) models [Goodfellow et al., 2014] and in particular Wasserstein GAN's [Arjovsky et al., 2017] to the problem of generating synthetic agents. Previous models [Farooq et al., 2013, Sun and Erath, 2015, Sun et al., 2018] have mainly worked on much smaller problems and excluded the job and sector information. Other models [Borysov et al., 2018], have successfully tackled high-dimensional problems in the context of population synthesis, however, jobs, sectors and the derived indirect commuting matrix were not modelled.

2 Literature review

Model-based population synthesis is a way to generate samples of a population. This is accomplished in two stages. First, by developing a flexible probabilistic model-based description of the underlying joint distribution, and secondly,

by adapting a sampling scheme where agents are drawn at random directly from the model. Recently, a number of applications to population synthesis have been proposed. [Sun and Erath, 2015] proposed a Bayesian network approach to population synthesis. The Bayesian network framework allows the modeller to define the relations between variables (conditionals) and infer a joint probability of the whole set of variables given these relations. The performance of the Bayesian network model is good compared to traditional approaches to population synthesis such as Iterative Proportional Filtering (IPF). [Sun et al., 2018] proposed a hierarchical mixture to population synthesis. This type of model assumes a mixture of multinomial random variables as the joint distribution of all the variables. Using the Expectation-Maximisation algorithm, inference on the parameters of the joint distribution including latent variables is performed. Just as in [Sun and Erath, 2015], the model performance is comparable to that of matrix fitting methods. One potential problem with [Sun and Erath, 2015, Sun et al., 2018] is that it remains unclear how to scale these models to high dimensions, when the number of modelled variables is high. Given the richness of contemporary surveys, we would like to synthesize population on a very detailed level. One solution to this problem is proposed in [Borysov et al., 2018]. The authors use a Variational Autoencoder (VAE) to do population synthesis. The VAE is a deep generative model which allows sampling from a learned joint distribution of the input data (see more in Section 3.1). This model scales to higher dimensions as opposed to the previous models and it is also considered in this paper as a benchmark.

3 Methodology

3.1 Variational Autoencoders

Variational Autoencoder, as presented in [Kingma and Welling, 2014], is an artificial neural network-based generative model from deep learning which aims to learn the joint distribution $p(x)$. It does so by the means of two neural networks trained simultaneously, the recognition network (also called inference network or "encoder") which takes the original data x as an input (in the present case, observations from the transport diaries) and outputs parameters of a prior distribution family (generally Gaussian) μ and σ of $q(z|x) = N(\mu, \sigma)$. The second network referred to as "decoder" takes a draw from the distribution with those parameters and tries to output an observation $\hat{x}$ as similar as possible to the original input. The networks are trained to minimize the Evidence Lower BOund (ELBO):

$$ELBO = E_{q_{\phi(z|x)}}[logp_\theta(x|z)] - D_{KL}(q_\phi(z|x)||p_\theta(z)).$$

The first term in the ELBO is called the reconstruction error and it measures how different is a generated sample (using the parameters from the prior distribution) from the real sample. The second term is the Kullback-Leibler divergence between the prior distribution and the posterior distribution. It can

also be viewed as a regularization term on the learned posterior distribution. Having been trained, the VAE is capable of generating new samples by sampling from the latent variable prior and transforming it using the decoder. For a more detailed explanation of this methodology, refer to the original paper [Kingma and Welling, 2014] or a more high-level explanation in the context of population synthesis in [Borysov et al., 2018].

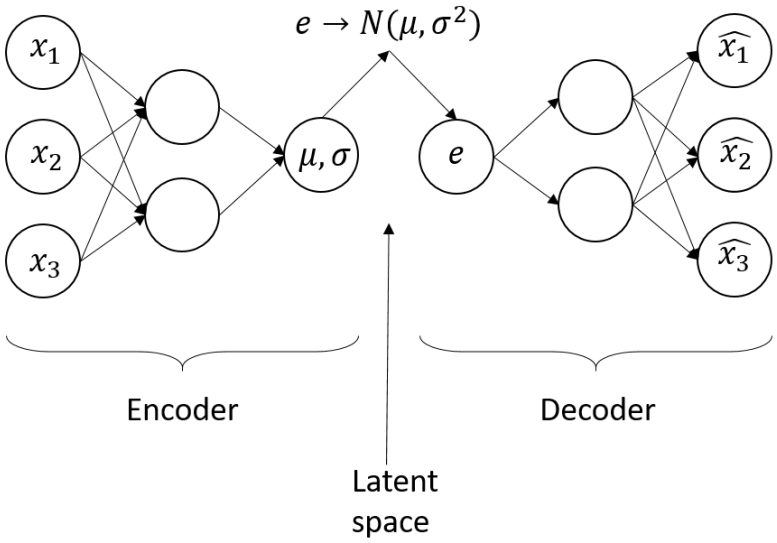


Figure 1: Variational Autoencoder

3.2 Generative Adversarial Networks

Generative Adversarial Network (GANs) [Goodfellow et al., 2014] is a generative model developed within the deep learning framework. This model is trained using the adversarial principle. In its basic form, GAN consists of two artificial neural networks. The first one is the generator G, which creates samples by transforming another variable z sampled from a prior distribution (usually the standard normal) with the objective to 'fool' the second network, the discriminator D. The discriminator has the objective of classifying real samples (from the original data set) and "fake" samples (samples from the generator) to determine whether they are real or not. Figure 2 depicts the training process. The losses of the networks are:

$$L_G = \log(1 - D(G(z)))$$

$$L_D = -(log(D(x)) + log(1 - D(G(z))))$$

As expected from the adversarial setting, the generator loss is inversely proportional to the outcome of the discriminator.

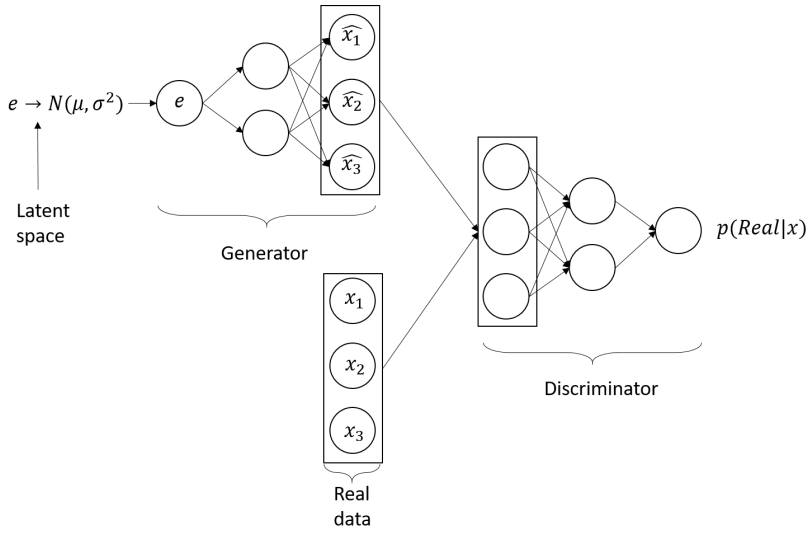


Figure 2: Generative Adversarial Network

GANs have a number of advantages, especially when applied to the image generation task, and offer a solid alternative to the Variational Autoencoder (VAE). However, this comes with the cost of some problems that are not present on VAEs. Mode collapse (also known as the "Helvetica scenario") is a problem that arises when the generator learns to reproduce almost exactly samples from the training data. Even though the training proceeds as expected, meaning that the generator has low loss, this behaviour is undesirable since we would like to have diverse enough samples from the generator. The deep learning literature has offered several solutions to this problem with different rates of success. One of these solutions is the Wasserstein GAN (WGAN) [Arjovsky et al., 2017] which uses the Wasserstein or Earth Mover's Distance (EMD) instead of the cross-entropy commonly used as a loss function to train the model.

The WGAN model is used in this paper. The important change being made with respect to the basic GAN setting is that we minimize the Wasserstein loss between two distributions: the generator distribution P_θ and the real distribution P_r. For practical matters, the losses of the generator and the discriminator networks are often represented in the following form:

$$L_G = -D(G(z))$$

$$L_D = -(D(x) - D(G(z)))$$

The intuition behind the Wasserstein distance is depicted in Figure 3.

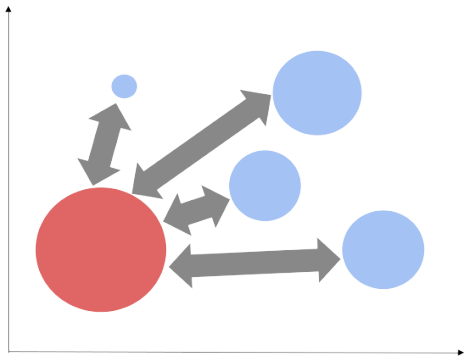


Figure 3: Sample Wasserstein loss problem. Given the two distributions, blue and red, the Wasserstein loss is the weighted minimum distance needed to convert one distribution into the other.

4 Data

This study is based on the data from the Danish National Travel Survey (TU) [DTU, 2017]. It contains 156.248 observations from 2006 until 2018. The data sum up to 459.572 trips and represent trips for all Danish citizens. The variables include socio-economic characteristics, such as income, number of persons in household; variables related to trips, such as municipality of origin, total time spent on mode, etc. Additionally, the data was further enriched by the job sector information by using the data from Statistic Denmark.

Data pre-processing included (i) transforming numerical variables into categorical variables which can take one of the 5 values based on the corresponding quantiles, (ii) removing the variables with more than 20% of missing values and (iii) merging of jobs locations and sectors from Statistics Denmark. The data set was divided in 3 folds: training (25%), validation (25%) and test (50%). Following the standard machine learning methodology, the training set is used to fit the model parameters, while the hyper-parameters are chosen based on the performance on the validation set. Finally, the best-performing model is evaluated using the test set. After the pre-processing, the data consisted 67 variables with 18.688 observations for training, 18.688 observations for validation, and 37.376 samples for test.

5 Results

Comparison of distributions and specifically of deep generative models is still an active area of research in the deep learning literature. [Theis et al., 2016] argue that the evaluation of generative models should be designed specifically for each task. Particularly, for the population synthesis generation, there is no clear way to evaluate the generated samples. Here, for instance, possible options can include comparison of the statistical moments or log-likelihood per-

formance. In the previous population synthesis literature, several measures have been tested, such as correlation coefficient (Corr), coefficient of determination (R^2), root mean squared error (RMSE) and its snadardised version (SRMSE). In this paper, we will focus on these measures calculated on different subsets of variables, i.e., the joint distribution of 2 or 3 variables with other variables being marginalised. We resort to these low-dimensional projections because of the extreme sparsity of the data, since the modelled agents are 762-dimensional within a 'one-hot' representation of the categorical variables. Some of the results of the WGAN can be seen in Figure 4 where they are compared side-by-side with the performance of the VAE.

In Figure 4 and Figure 5, we present the goodness-of-fit of the models. Figure 4 compares the joint distributions of the model with that of the test population. In order to create this figure, we generate a pool of samples from the trained model, we pick a subset of variables and, for all the combinations of categories of this subset, we estimate the probability of this combination. We do this process for both the sampled pool of individuals and for the test pool of individuals. Furthermore, we plot the predicted distribution against the test distributions. Ideally, the points in the scatter plot would be close to the 45 degree line. For the selected variables, we can see that the WGAN has better performance than the VAE in all settings except for the municipality of destination and sector. However, to the authors best judgment, the difference in performance seems to be small. Figure 5 shows, for some variables, the probability of belonging to a certain category for a certain variable. The probabilities of the observations from the model are plotted side-by-side to those of the test. In these figures, it seems that the VAE approximates the marginal distributions slightly better than the WGAN does for the three selected variables.

6 Conclusion

Population synthesis is a fundamental task when modelling transportation demand. If not properly modelled, it will lead to errors propagating in subsequent model stages. Different approaches have been proposed in the past with some success. However, some of the problems related to these frameworks, such as working with sparse matrices or high dimensions in combination with small data set has led to new ways of doing population synthesis. One type of the recently explored models is the probabilistic model where the initial sample is considered as a realization from a real joint distribution and the aim of the modeller is to find this distribution.

In this paper, we proposed a methodology to solve some of these problems. The models come from deep learning—an active research field studying artificial neural networks applied to high-dimensional data. Working with the data from Denmark, we used the Generative Adversarial Networks (GAN) approach to synthesise population. We compared it to the Variational Autoencoder (VAE), another popular deep generative modelling approach.

Deep generative models represent a powerful set of tools that can be applied

to the transportation problems and more specifically on population synthesis. Some issues when the deep generative models applied to such kinds of problems remain unsolved. It is unclear how to rigorously test the performance of these models. It is also important to check the consistency of the generated data with respect to the possible agents. This means, checking whether the generated population does not contain any logical inconsistencies, for instance, a kid with a driving license. Another interesting problem that remains unsolved is how to align generated populations with future population targets created by demographic models.

References

[Abadi et al., 2015] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

[Arjovsky et al., 2017] Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein generative adversarial networks. In Precup, D. and Teh, Y. W., editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 214–223, International Convention Centre, Sydney, Australia. PMLR.

[Beckman et al., 1996] Beckman, R. J., Baggerly, K. A., and McKay, M. D. (1996). Creating synthetic baseline populations. *Transportation Research Part A: Policy and Practice*, 30(6):415–429.

[Borysov et al., 2018] Borysov, S., Rich, J., and Pereira, F. (2018). Scalable population synthesis with deep generative modeling [arxiv]. *ArXiv*.

[Deming and Stephan, 1940] Deming, W. E. and Stephan, F. F. (1940). On a least squares adjustment of a sampled frequency table when the expected marginal totals are known. *The Annals of Mathematical Statistics*, 11(4):427–444.

[DTU, 2017] DTU, C. F. T. A. F. D. T. (2017). Danish national travel survey, datasets.

[Farooq et al., 2013] Farooq, B., Bierlaire, M., Hurtubia, R., and Flötteröd, G. (2013). Simulation based population synthesis. *Transportation Research Part B: Methodological*, 58:243–263.

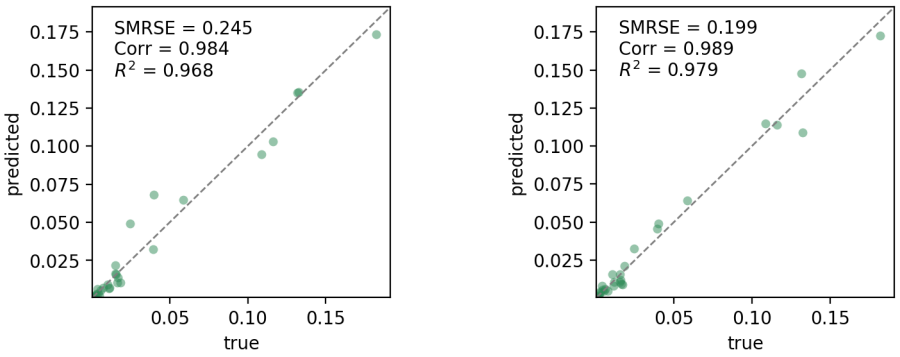
[Goodfellow et al., 2014] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc.

[Kingma and Welling, 2014] Kingma, D. P. and Welling, M. (2014). Auto-Encoding Variational Bayes. *arXiv e-prints*, page arXiv:1312.6114.

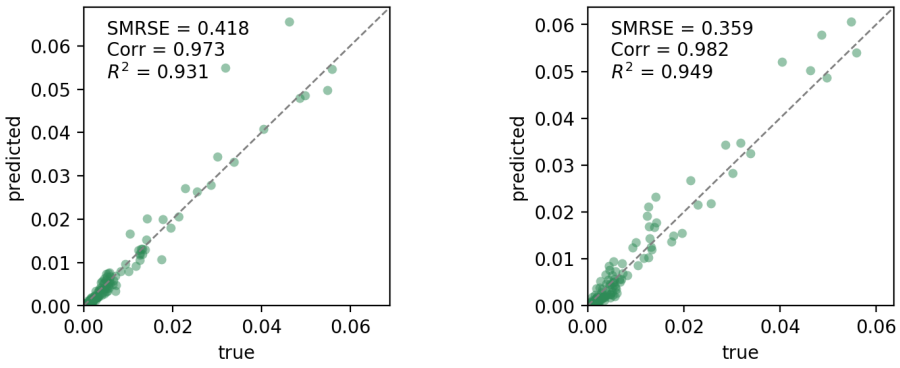
[Sun and Erath, 2015] Sun, L. and Erath, A. (2015). A bayesian network approach for population synthesis. *Transportation Research Part C: Emerging Technologies*, 61:49–62.

[Sun et al., 2018] Sun, L., Erath, A., and Cai, M. (2018). A hierarchical mixture modeling framework for population synthesis. *Transportation Research Part B: Methodological*, 114:199–212.

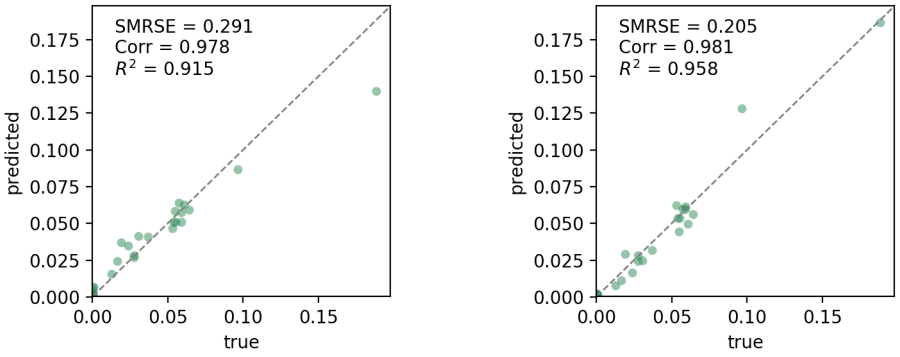
[Theis et al., 2016] Theis, L., van den Oord, A., and Bethge, M. (2016). A note on the evaluation of generative models. In *International Conference on Learning Representations*.



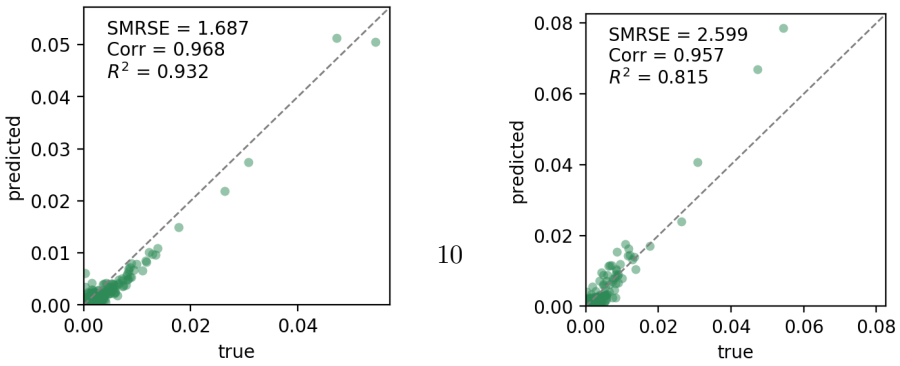
(a) Gender, Popsocio, Education



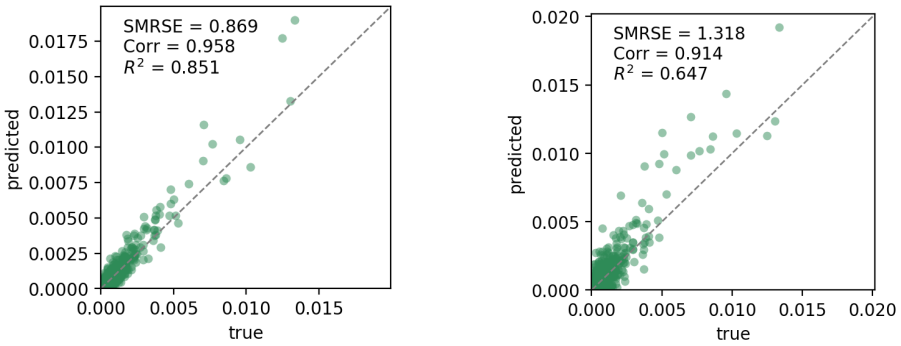
(b) Gender, Popsocio, Sector



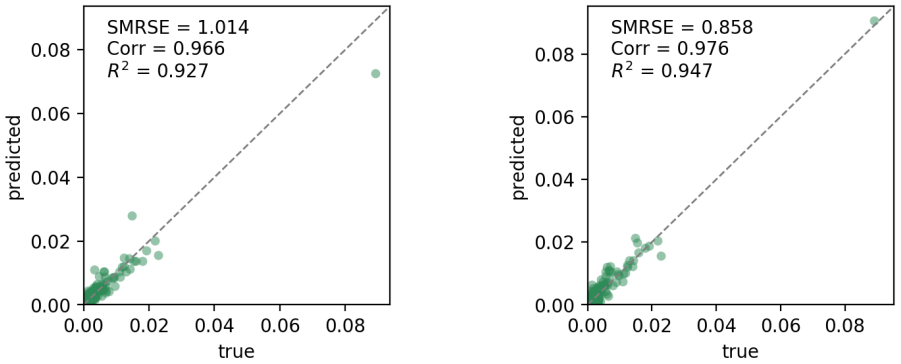
(c) Income respondent, Age



(d) Municipality of origin, Municipality of destination

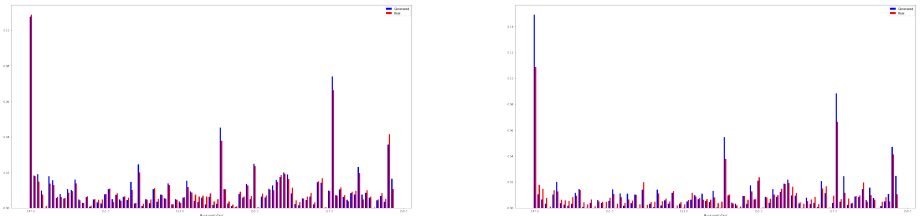


(e) Municipality of destination, Sector

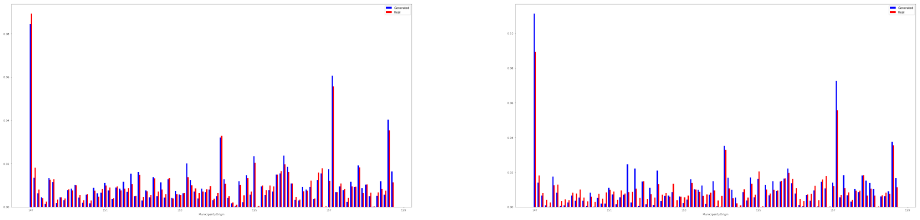


(f) Sector, Income respondent, Education

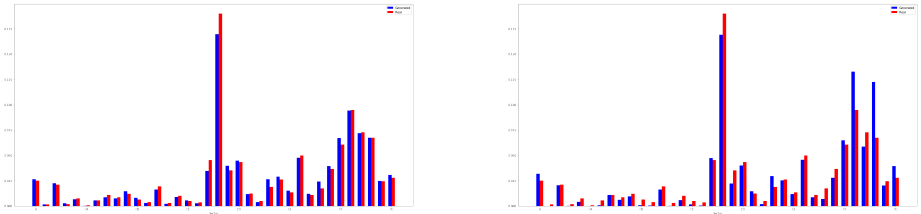
Figure 4: Joint distributions for different subset of variables generated by WGAN (right) and VAE (left) compared to the joint distribution of the test data.



(a) Municipality of destination



(b) Municipality of origin



(c) Sector

Figure 5: Marginal distributions generated by WGAN (right) and VAE (left) compared to the marginal distributions of the test data.

Intermediate dimensions	1024
Latent dimension	100
Number of hidden layers	1
Beta	7.21e(-03)
Categorical loss weight	8.70
Learn rate	7.51e(-03)
Epochs	100
Batch size	256
Optimizer	Adam
Batch normalization	yes
Early stop	yes

Table 1: Architecture of the VAE

Intermediate dimensions	1024
Latent dimension	100
Number of hidden layers	1
Beta	7.21e(-03)
Categorical loss weight	8.70
Learn rate	7.51e(-04)
Epochs	10000
Batch size	256
Optimizer	RMS Prop
Number of critic updates	5
Clip value	0.01

Table 2: Architecture of the WGAN, notice that the generator and the critic (discriminator) were created symmetrically