CAN AUTO-ENCODERS HELP WITH FILLING MISSING DATA?

Marek Śmieja
Jagiellonian University
Kraków, Poland
marek.smieja@uj.edu.pl

Maciej Kołomycki
Cracow University
of Technology,
Kraków, Poland
maciej.kolomycki@pk.edu.pl

Łukasz Struski
Jagiellonian University
Kraków, Poland
lukasz.struski@uj.edu.pl

Mateusz Juda
Jagiellonian University
Kraków, Poland
mateusz.juda@uj.edu.pl

Mário A. T. Figueiredo
Instituto de Telecomunicaes,
Instituto Superior Tcnico,
Universidade de Lisboa, Portugal
mario.figueiredo@tecnico.ulisboa.pt

ABSTRACT

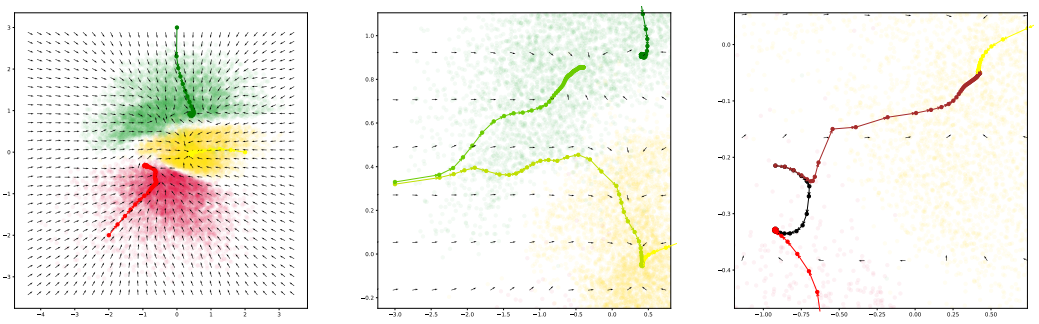
This paper introduces an approach to filling in missing data based on deep auto-encoder models, adequate to high-dimensional data exhibiting complex dependencies, such as images. The method exploits the properties of auto-encoders' vector fields, which allows to approximate the gradient of the log-density from its reconstruction error, based on which we propose a projected gradient ascent algorithm to obtain the conditionally most probable estimate of the missing values. Experiments performed on benchmark datasets show that imputations produced by our model are sharp and realistic.

1 INTRODUCTION

Filling in missing data is an important problem in machine learning and data analysis, especially when dealing with real-world data (Luo et al., 2018; Hwang et al., 2019; Camino et al., 2019). A classical approach relies on density estimation, e.g., using a *Gaussian mixture models* (GMM) (Titterington & Sedransk, 1989). Using the estimated density, the missing values of each observation are then replaced either by samples or by maximizers of the corresponding conditional density, given the observed ones. This approach tends to perform well, as long as the density model is expressive enough for the data at hand and is accurately estimated.

Although the use of a shallow density model, e.g. GMM, may allow obtaining the conditional density analytically, as well as easily sampling from it or finding its maximizers, such a model may be unable to efficiently describe complex dependencies in real data, such as images (Śmieja et al., 2018). While deep generative models, e.g. *generative adversarial networks* (GAN) (Goodfellow et al., 2014) or *variational autoencoders* (VAE) (Kingma & Welling., 2014), are sufficiently expressive, it may be impossible to obtain or maximize the corresponding conditional densities of missing values given the observed ones. Consequently, using deep generative models for density-based imputation is a challenging task (Mattei & Frellsen, 2018).

In this paper, we exploit the dynamics of auto-encoders' reconstruction function. Based on theoretical results presented by Alain & Bengio (2014), the reconstruction error of a *denoising auto-encoder* (DAE) (Vincent, 2011) provides an approximate expression for the gradient of the logarithm of the probability density function (pdf), which is (implicitly) estimated from data. We adapt that approach



(a) Trajectories of points selected from the latent space.

(b) Splits of the basins of attractions.

(c) An attractor in a low density region.

Figure 1: Example of a latent space trajectoris following the latent vector field u, for a VAE trained on the MNIST dataset (classes 0, 1, 2). Dots represent latent representation of the training examples: 0-red ,1 -green, 2-yellow).

in the context of incomplete data to maximize the conditional density of the missing values, given the observed ones. The conditionally most probable values are found as the attractors of the iterated reconstruction function.

We instantiate the proposed approach using two types of auto-encoders: DAE and WAE (Wasserstein auto-encoders) (Tolstikhin et al., 2017). Experiments with the MNIST and Fashion-MNIST datasets show that both models provide very good results (the best-looking completions in the case of images with missing pixels).

2 AUTO-ENCODER'S DYNAMICAL SYSTEM

To motivate our approach to missing data imputation, we recall known facts regarding the vector fields generated by auto-encoders' reconstruction functions.

An auto-encoder may be viewed as composition of two maps, an *encoder* $f : \mathbb{R}^d \to Z$ and a *decoder* $g : Z \to \mathbb{R}^d$, such that $Z \subset \mathbb{R}^l$ is a *latent space* and the *reconstruction* function $r := g \circ f$ is close to identity, *i.e.*, $r(x) \approx x$. Since auto-encoders, in general, do not achieve perfect reconstructions, we can define an *auto-encoder vector field* $v : \mathbb{R}^d \to \mathbb{R}^d$ associated to the reconstruction function as $v(x) := r(x) - x$. Analogously, we also define an *auto-encoder latent vector field* $u : Z \to Z$ given by $u(z) := f(g(z)) - z$. A natural question arises: what is the structure of the dynamics generated by v and u.

The properties of the vector fields v for DAEs were studied and discussed in details by Alain & Bengio (2014). Namely the reconstruction error at some point $x \in \mathbb{R}^d$, produced by a DAE, is approximately equal to the gradient of the log-pdf computed at that point, in the low-noise limit, *i.e.*,

$$\frac{\partial \log p_X(x)}{\partial x} \approx \frac{r_\sigma(x) - x}{\sigma^2} = \frac{v_\sigma(x)}{\sigma^2}, \text{ as } \sigma \to 0, \tag{1}$$

where r_σ is the reconstruction function of the DAE at denoising level σ and v_σ is the corresponding vector field. Consequently, the point with the highest pdf can be found via gradient ascent (gradient flow, in the limit of infinitesimal steps) by exploiting this equality.

Analyzing the dynamics of the reconstruction error may be useful in verifying the quality of auto-encoders. Our intuition is that the iterated reconstruction function (and its counterpart in the latent space) should have a stationary point at an attractor. For an attractor, its basin of attraction should represent a subset of the space where the points with similar features.

As an example we consider a convolutional VAE for the MNIST dataset. We use latent space dimension $l = 2$ and train the model only for the digits 0, 1, and 2. For a point z in the latent space Z, we draw the latent trajectory generated by the auto-encoder, *i.e.*, $z_{i+1} := f(g(z_i))$. In most cases, we observe the behavior shown in Figure 1a: each trajectory travels through the latent

space and converges to a fixed point. However, a small perturbation of the starting point may cause a trajectory to converge to a different region, as illustrated in Figure 1b, where a data point for the digit 1 after several iteration is reconstructed as a digit 2. Yet another challenge arises if an attractor is located close to another basin of attraction; because of numerical instabilities, we may miss the attractor and be redirected to nearby region (see Figure 1c).

3 IMPUTATION MODEL

In this section, we show how to find attractors of iterated reconstruction functions in the context of missing data and, as a consequence, generate imputations with the highest local probability.

A point with missing data is denoted by a pair (x, J), where $x \in \mathbb{R}^d$ and $J \subset \{1, \ldots, d\}$ is the set of indices with missing values; for a fully-observed point, $J = \emptyset$. The question is: what is the "best" choice for filling the missing coordinates x_J (restriction of x to unobserved components)? In this paper, we tackle this problem in a classical probabilistic way: we choose the maximizer of the corresponding conditional pdf, given the observed variables $x_{\bar{J}}$, where $\bar{J} = \{1, ..., d\} \setminus J$ is the set of indices of the observed components of x.

To make the above statement more precise, let p_X be a pdf defined on $\mathbb{R}^d$, estimated from a dataset $X \in \mathbb{R}^{d \times n}$. Given a data point with missing components (x, J), assume that $J \neq \emptyset$, otherwise imputation is unnecessary, and $J \neq \{1, ..., d\}$, otherwise we do not have an imputation problem. The conditional pdf is given by Bayes law,

$$p(x_J | x_{\bar{J}}) = \frac{p(x_J, x_{\bar{J}})}{p(x_{\bar{J}})} = \frac{p_X(x)}{p(x_{\bar{J}})}, \tag{2}$$

because $x_{J \cup \bar{J}} = x \in \mathbb{R}^d$ (missing and observed). Since we are looking for the maximizer of this conditional pdf, the denominator is irrelevant:

$$\hat{x}_J = \arg \max_{x_J \in \mathbb{R}^{|J|}} p(x_J | x_{\bar{J}}) = \arg \max_{y \in \mathbb{R}^d : y_{\bar{J}} = x_{\bar{J}}} \log p_X(y). \tag{3}$$

In this work, we exploit the properties of auto-encoders' reconstruction function to seek the maximizer of the conditional density as defined in Equation 3. Based on the approximate formula for the gradient of log-pdf from Equation 1, we propose the following natural procedure:

1. Train an auto-encoder model on a dataset X.
2. Given a data point with missing values (x, J), randomly pick an initial filling $\hat{x}_J^0$ of the missing part x_J.
3. Iteratively update $\hat{x}_J$ using $\hat{x}_J^{t+1} = \hat{x}_J^t + h \, [r_\sigma(\hat{x}^t) - \hat{x}^t]_J$,

where h is a step size and $\hat{x}^t = (\hat{x}_J^t, x_{J'}) \in \mathbb{R}^d$ denotes a complete point where the observed components are fixed at the observed values and the missing ones are replaced by the current estimate. This procedure corresponds to moving on an (axes-aligned) affine subspace of dimension $\mathbb{R}^{|J|}$ of the data space $\mathbb{R}^d$ in a direction determined by the gradient of the log-density function. Because of the axes-aligned nature of the affine subspace, this coincides with a projected gradient ascent algorithm.

4 EXPERIMENTS

In this section, we experimentally assess the proposed model. We instantiate our procedure with a *denosing auto-encoder* (DAE) and a *Wasserstein auto-encoder* (WAE). We use architectures from the C.1. MNIST experiment reported by Tolstikhin et al. (2017). The step size for iterative filling of missing data is $h = 0.001$.

As a baseline, we consider a type of *context encoder* (CE) (Pathak et al., 2016), which fills in missing regions by minimizing the L_2 norm between input and output of the auto-encoder. The encoder of CE is composed of three convolutional layers, while decoder contains analogical transpose convolutions. Additionally, we used two typical imputation methods: (a) k-NN (Batista & Monard, 2002), which fills missing features with mean values of those features computed from the k nearest

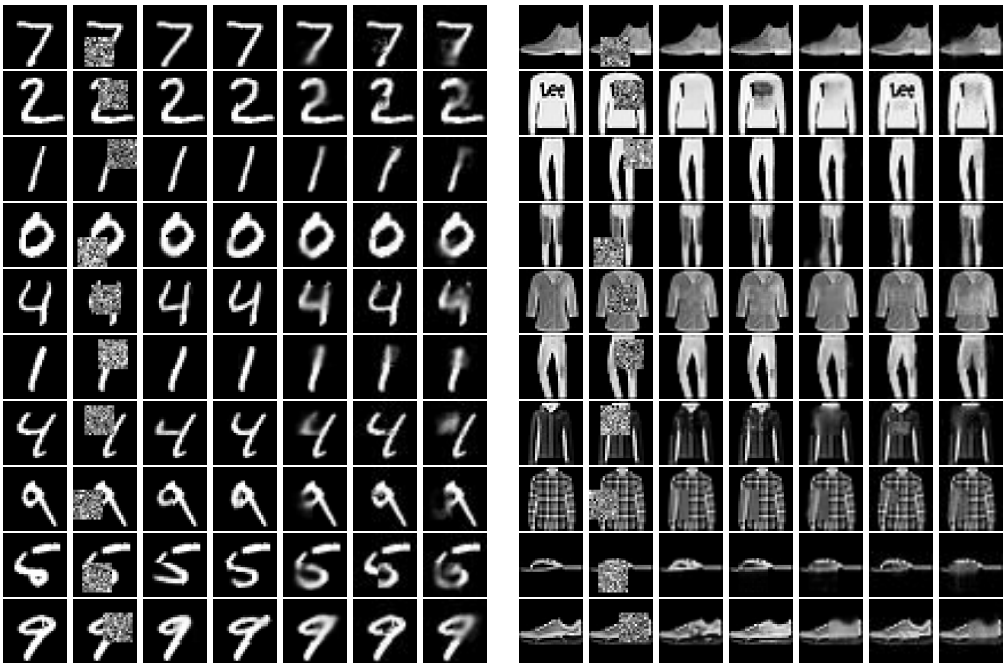


Figure 2: Reconstructions of partially incomplete images from MNIST (letf) and Fashion-MNIST (right) datasets. From left: (1) original image, (2) initial random filling of missing region, and imputations using (3) DAE dynamics, (4) WAE dynamics, (5) CE (6) k-NN, (7) MICE.

training samples (we used k = 5); (b) MICE (Buuren & Groothuis-Oudshoorn, 2010; Azur et al., 2011), where several imputations are drawn from the conditional pdf using Markov chain Monte Carlo sampling.

We consider two standard datasets: MNIST (LeCun et al., 1998) and Fashion-MNIST (Xiao et al., 2017). For each test image of the size 28×28, we drop a square patch of size 13×13, at a random location (uniformly sampled for each image).

Figure 2 presents sample results obtained by the different methods considered. In this experiment, the DAE and WAE gave the best-looking imputations; they are sharp and look visually plausible. One can observe that WAE usually produces more details (see 2nd and 9th rows for Fashion-MNIST), however this is not always the case (see 10th row for Fashion-MNIST, where imputation returned by DAE is more realistic). On the other hand, the imputations returned by CE, k-NN, and MICE are often blurry (especially so for CE and MICE).

To provide a quantitative assessment of the methods, we report *mean square error* (MSE) values in Table 1. Although the CE imputations are quite blurry, this method achieves the lowest MSE on both datasets. This means that, although the filled-in pixels agree with the original images on average, the results do not have to be realistic. As can be seen, WAE also obtained very low errors while maintaining sharp images, while DAE gave only slightly worse results for Fashion-MNIST. It is not surprising that CE provides the lowest MSE, because its cost function directly focuses on this goal. On the other hand, our method tries to fit the hole with the most probable values, yielding sharp and realistic images, which may not match exactly with the true image.

One possible explanation for why the WAE performed slightly better than the DAE could be connected with its generative nature. The WAE tries to fit the encoded data to a Gaussian, which reduces holes in the latent space.

5 Conclusion

In this paper, we investigated auto-encoder vector fields and proposed a strategy for filling in missing values. Our approach does not require any specialized network architectures and training proce-

Table 1: Mean square error of imputations.

Dataset	DAE	WAE	CE	k-NN	MICE
MNIST	0.0872	0.0864	0.0588	0.0879	0.0811
Fashion-MNIST	0.0328	0.0286	0.0243	0.0324	0.0302

dures. Given any AE model, we traverse through its vector field to its attractors, which are elements with the highest probability (in some local neighborhood). Experiments showed that this cheap procedure leads to realistically-looking imputations.

ACKNOWLEDGEMENT

We thank Jacek Tabor and Przemysław Spurek for inspiring discussions and their help at the initial stage of this work.

This work was partially supported by the National Science Centre (Poland) grant no. 2014/14/A/ST1/00453, 2015/19/D/ST6/01215, 2017/25/B/ST6/01271, and 2018/31/B/ST6/00993.

REFERENCES

G. Alain and Y. Bengio. What regularized auto-encoders learn from the data-generating distribution. *The Journal of Machine Learning Research*, 15(1):3563–3593, 2014.

M. Azur, E. Stuart, C. Frangakis, and P. Leaf. Multiple imputation by chained equations: what is it and how does it work? *International journal of Methods in Psychiatric Research*, 20:40–49, 2011.

G. Batista and M. Monard. A study of k-nearest neighbour as an imputation method. *Frontiers in Artificial Intelligence and Applications*, 97:251–260, 2002.

S. Buuren and K. Groothuis-Oudshoorn. mice: Multivariate imputation by chained equations in r. *Journal of Statistical Software*, pp. 1–68, 2010.

R. Camino, C. Hammerschmidt, and R. State. Improving missing data imputation with deep generative models. *arXiv preprint arXiv:1902.10666*, 2019.

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, , and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pp. 2672–2680, 2014.

U. Hwang, D. Jung, and S. Yoon. Hexagan: Generative adversarial nets for real world classification. *arXiv preprint arXiv:1902.09913*, 2019.

D. Kingma and M. Welling. Auto-encoding variational Bayes. In *Proceedings of the 2nd International Conference on Learning Representations*, 2014.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86:2278–2324, 1998.

Y. Luo, X. Cai, Y. Zhang, J. Xu, and Y. Xiaojie. Multivariate time series imputation with generative adversarial networks. In *Advances in Neural Information Processing Systems*, pp. 1596–1607, 2018.

P. Mattei and J. Frellsen. Miwae: Deep generative modelling and imputation of incomplete data. *arXiv preprint arXiv:1812.02633*, 2018.

D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2536–2544, 2016.

M. Śmieja, Ł. Struski, J. Tabor, B. Zieliński, and P. Spurek. Processing of missing data by neural networks. In *Advances in Neural Information Processing Systems*, pp. 2719–2729, 2018.

D. Titterington and J. Sedransk. Imputation of missing values using density estimation. *Statistics & Probability Letters*, 9(5):411–418, 1989.

I. Tolstikhin, O. Bousquet, S. Gelly, and B. Schölkopf. Wasserstein auto-encoders. arXiv:1711.01558, 2017.

P. Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7), 2011.

H. Xiao, K. Rasul, and R. Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.