missIWAE: Deep Generative Modelling and Imputation of Incomplete Data

Pierre-Alexandre Mattei
Department of Computer Science
IT University of Copenhagen
pima@itu.dk

Jes Frellsen
Department of Computer Science
IT University of Copenhagen
jefr@itu.dk

Abstract

We present a simple technique to train deep latent variable models (DLVMs) when the training set contains missing data. Our approach is based on the importance-weighted autoencoder (IWAE) of Burda et al. (2016), and also allows single or multiple imputation of the incomplete data set. We illustrate it by training a convolutional DLVM on a static binarisation of MNIST that contains 50% of missing data. Leveraging mutiple imputations, we train a convolutional network that classifies these incomplete digits as well as complete ones.

1 Training deep generative models with missing data

We start with some i.i.d. data stored in a matrix $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)^T \in \mathcal{X}^n$. We assume that p different features are present in the data, leading to the decomposition $\mathcal{X} = \mathcal{X}_1 \times \ldots \times \mathcal{X}_p$ (for example, if all features are continuous, $\mathcal{X} = \mathbb{R}^p$). When some data are missing, we split each sample $i \in \{1, \ldots, n\}$ into the observed features $\mathbf{x}_i^{\text{o}}$ and the missing features $\mathbf{x}_i^{\text{m}}$. The indices of the missing features are stored in binary vectors $\mathbf{m}_i \in \{0, 1\}^p$ such that $m_{ij} = 0$ if feature j is observed for sample i, and $m_{ij} = 1$ if feature j is missing.

We wish to explain these potentially high-dimensional data using some latent variables $\mathbf{Z} = (\mathbf{z}_1, \ldots, \mathbf{z}_n)^T \in \mathbb{R}^{n \times d}$. While the dimension d of the latent space will often be smaller than the dimension of the input space $\mathcal{X}$, this will not always be the case.

1.1 Deep Latent Variable Models

DLVMs (Rezende et al., 2014; Kingma and Welling, 2014) assume that $(\mathbf{x}_i, \mathbf{z}_i)_{i \leq n}$ are driven by the following generative model:

$$\begin{cases} \mathbf{z} \sim p(\mathbf{z}) \\ p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}) = \Phi(\mathbf{x}|f_{\boldsymbol{\theta}}(\mathbf{z})), \end{cases} \tag{1}$$

where $(\Phi(\cdot|\boldsymbol{\eta}))_{\boldsymbol{\eta} \in H}$ is a parametric family of distributions over $\mathcal{X}$ called the *observation model*. Often, it is a very simple family such as the Gaussian distribution if $\mathcal{X}$ is continuous, or products of multinomial distributions if $\mathcal{X}$ is discrete. The function $f_{\boldsymbol{\theta}} : \mathbb{R}^d \to H$ is called the *decoder* (or the *generative network*), and is parametrised by a deep neural network whose weights are stored in $\boldsymbol{\theta} \in \boldsymbol{\Theta}$. Here, we will make the general assumption that the observation model $(\Phi(\cdot|\boldsymbol{\eta}))_{\boldsymbol{\eta} \in H}$ is such that all its marginal and conditional distributions are available in closed-form, which is true for commonly used observation models like the Gaussian or the Bernoulli ones. This assumption will conveniently guarantee that the quantity $p_{\boldsymbol{\theta}}(\mathbf{x}^{\text{o}}|\mathbf{z})$ is easy to compute.

DLVMs are usually trained using approximate maximum likelihood techniques that maximise lower bounds of the log-likelihood function. In our case, the likelihood of the *observed* data $\mathbf{x}_1^{\text{o}}, \ldots, \mathbf{x}_n^{\text{o}}$ is

Third workshop on Bayesian Deep Learning (NeurIPS 2018), Montréal, Canada.

equal to

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log p_{\boldsymbol{\theta}}(\mathbf{x}_i^{\text{o}}) = \sum_{i=1}^{n} \log \int p_{\boldsymbol{\theta}}(\mathbf{x}_i^{\text{o}}|\mathbf{z}) p(\mathbf{z}) d\mathbf{z}. \tag{2}$$

Under the assumption that the data are missing-at-random, maximising $\ell(\boldsymbol{\theta})$ is a sound inference procedure. Unfortunately, the integrals involved in this expression make direct maximum likelihood impractical. However, it is possible to derive tractable tight lower bounds of $\ell(\boldsymbol{\theta})$ whose maximisation is much easier. More precisely, we will base our inference strategy on the importance-weighted autoencoder (IWAE) of Burda et al. (2016), which combines ideas from amortised variational inference and importance sampling.

To build a lower bound using amortised variational inference, we will define a parametrised conditional distribution $q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x}^{\text{o}})$ called the *variational distribution* that will *play the role of a proposal distribution close to the intractable posterior* $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}^{\text{o}})$. Specifically this conditional distribution will be defined as $q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x}^{\text{o}}) = \Psi(\mathbf{z}|g_{\boldsymbol{\gamma}}(\iota(\mathbf{x}^{\text{o}}))$, where ι is an *imputation function chosen beforehand* that transforms $\mathbf{x}^{\text{o}}$ and $\mathbf{m}$ into a complete input vector $\iota(\mathbf{x}^{\text{o}}) \in \mathcal{X}$ such that $\iota(\mathbf{x}^{\text{o}})^{\text{o}} = \mathbf{x}^{\text{o}}$. The set $(\Psi(\cdot|\boldsymbol{\kappa}))_{\boldsymbol{\kappa}\in\mathcal{K}}$ is a parametric family of simple distributions over $\mathbb{R}^d$, called the *variational family*, and the function $g_{\boldsymbol{\gamma}} : \mathcal{X} \to \mathcal{K}$, called the *inference network* or the *encoder*, is parametrised by a deep neural network whose weights are stored in $\boldsymbol{\gamma} \in \boldsymbol{\Gamma}$. Its role will be to transform each data point into the parameters of Ψ.

Following the steps of Burda et al. (2016), we can use the distribution $q_{\boldsymbol{\gamma}}$ to build approachable stochastic lower bounds of $\ell(\boldsymbol{\theta})$. More specifically, given $K \in \mathbb{N}^*$, we define the *missing data importance-weighted autoencoder (missIWAE) bound*

$$\mathcal{L}_K(\boldsymbol{\theta}, \boldsymbol{\gamma}) = \sum_{i=1}^{n} \mathbb{E}_{\mathbf{z}_{i1},\ldots,\mathbf{z}_{iK}\sim q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x}_i^{\text{o}})} \left[\log \frac{1}{K} \sum_{k=1}^{K} \frac{p_{\boldsymbol{\theta}}(\mathbf{x}_i^{\text{o}}|\mathbf{z}_{ik}) p(\mathbf{z}_{ik})}{q_{\boldsymbol{\gamma}}(\mathbf{z}_{ik}|\mathbf{x}_i^{\text{o}})} \right]. \tag{3}$$

When $K = 1$, the bound resembles the variational autoencoder (VAE, Kingma and Welling, 2014) objective, so we call this bound the *missVAE bound*. In very interesting concurrent work, Nazabal et al. (2018) derived independently the missVAE bound. They also discuss efficient strategies to design observation models that handle heterogeneous data sets.

This quantity is exactly the expected value of what we would get if we were to estimate the log-likelihood by approximating the integrals $p_{\boldsymbol{\theta}}(\mathbf{x}_1), \ldots, p_{\boldsymbol{\theta}}(\mathbf{x}_n)$ present in Eq. (2) using importance sampling, with proposal distributions $q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x}_1^{\text{o}}), \ldots, q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x}_n^{\text{o}})$. Regarding these importance sampling problems, it follows from a classical result of Monte Carlo integration (see e.g. Robert and Casella 2004, Section 3.3.2) that the optimal (in the sense of minimising the variance of the estimate) proposal distributions would exactly be the posterior distributions $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}_1^{\text{o}}), \ldots, p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}_n^{\text{o}})$. For this reason, for all $i \in \{1, \ldots, n\}$, we may interpret (and use) $q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x}_i^{\text{o}})$ as an approximation of the posterior.

Jensen's inequality ensures that $\mathcal{L}_K(\boldsymbol{\theta}, \boldsymbol{\gamma}) \leq \ell(\theta)$, which means that $\mathcal{L}_K$ is indeed a lower bound of the likelihood of the incomplete data set. Rather than optimising the intractable likelihood $\ell(\theta)$, we will maximise $\mathcal{L}_K(\boldsymbol{\theta}, \boldsymbol{\gamma})$ with respect to both $\boldsymbol{\theta}$ and $\boldsymbol{\gamma}$.

1.2 Properties of the missIWAE bound

The use of the imputation function ι, which fills in the missing values in each data point in order to feed it to the encoder, is the main originality of the missIWAE bound. Intuitively, it would be desirable to use an accurate imputation function. However, thanks to the properties of importance sampling, *using a very rough imputation is acceptable*, when K is large. Indeed, a direct consequence of the Theorem 1 of Burda et al. (2016) is that when the posterior distributions have lighter tails than their variational counterparts: $\mathcal{L}_1(\boldsymbol{\theta}, \boldsymbol{\gamma}) \leq \mathcal{L}_2(\boldsymbol{\theta}, \boldsymbol{\gamma}) \leq \ldots \leq \mathcal{L}_K(\boldsymbol{\theta}, \boldsymbol{\gamma}) \xrightarrow[K\to\infty]{} \ell(\boldsymbol{\theta})$.

Consequently, we suggest that the function ι *can simply be the zero imputation function*, which replaces missing values by zeroes. More complex imputations can of course be used; it would also be possible to use a parametrised imputation function that could be learned by maximising $\mathcal{L}_K$.

Regarding the variational family, we choose to limit our experiments to the family of Gaussian distributions with diagonal covariances, similarly to the original IWAE of Burda et al. (2016). However, it is worth noticing that our missIWAE bound could be used for any reparametrisable variational family, like the ones listed by Figurnov et al. (2018). Moreover, an interesting alternative

to Gaussians would be to consider elliptical distributions, as recently proposed by Domke and Sheldon (2018).

In general, the encoder can simply be a fully connected network, as in Burda et al. (2016). However, when dealing with images or sounds, it will often be more reasonable to use a convolutional network, as in Salimans et al. (2017). When the data are sequential, recurrent networks are also an option (see e.g. Bowman et al. 2016; Gómez-Bombarelli et al. 2018).

2 Missing data imputation

We assume in this section that we are given a data point $\mathbf{x} \in \mathcal{X}$ composed of some observed features $\mathbf{x}^{\text{o}}$ and some missing data $\mathbf{x}^{\text{m}}$. This data point can be taken from the training set, or from another incomplete data set sampled from the same distribution.

Since we have already learned a generative model $p_{\boldsymbol{\theta}}$, a good way of imputing $\mathbf{x}^{\text{m}}$ would be to sample according to its conditional distribution

$$p_{\boldsymbol{\theta}}(\mathbf{x}^{\text{m}}|\mathbf{x}^{\text{o}}) = \int p_{\boldsymbol{\theta}}(\mathbf{x}^{\text{m}}|\mathbf{x}^{\text{o}}, \mathbf{z}) p(\mathbf{z}|\mathbf{x}^{\text{o}}) d\mathbf{z}. \tag{4}$$

The nonlinearity of the decoder makes this conditional distribution hard to assess. However, it is possible to build a Metropolis-within-Gibbs sampler whose stationary distribution is exactly $p_{\boldsymbol{\theta}}(\mathbf{x}^{\text{m}}|\mathbf{x}^{\text{o}})$ (Mattei and Frellsen, 2018). The Metropolis-within-Gibbs sampler of Mattei and Frellsen (2018) was designed to leverage DLVMs trained on complete data, and critically requires the availability of a good approximation of the posterior distribution of the complete data $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})$. While training a VAE or an IWAE will provide an approximation of $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})$ via the inference network, using the missIWAE bound rather provides an approximation of $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}^{\text{o}})$. Therefore, the scheme of Mattei and Frellsen (2018) seem unfit for our purposes. We will therefore derive a new imputation technique compatible with a DLVM trained using the missIWAE bound. The main idea is to leverage the fact that $q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x}^{\text{o}}) \approx p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}^{\text{o}})$.

Single Imputation with missIWAE. Let us first focus on the *single imputation problem*: finding a single imputation $\hat{\mathbf{x}}^{\text{m}}$ that is close to the true $\mathbf{x}^{\text{m}}$. If the data are continuous and the ℓ_2 norm is a relevant error metric, then the optimal decision-theoretic choice would be $\hat{\mathbf{x}}^{\text{m}} = \mathbb{E}[\mathbf{x}^{\text{m}}|\mathbf{x}^{\text{o}}]$, which is likely to be intractable for the same reasons $p_{\boldsymbol{\theta}}(\mathbf{x}^{\text{m}}|\mathbf{x}^{\text{o}})$ is. We can actually give a recipe to estimate the more general quantity $\mathbb{E}[h(\mathbf{x}^{\text{m}})|\mathbf{x}^{\text{o}}]$, where $h(\mathbf{x}^{\text{m}})$ is any absolutely integrable function of $\mathbf{x}^{\text{m}}$. Indeed, this integral can be estimated using self-normalised importance sampling with the proposal distribution $p_{\boldsymbol{\theta}}(\mathbf{x}^{\text{m}}|\mathbf{x}^{\text{o}}, \mathbf{z}) q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x}^{\text{o}})$, leading to the estimate $\mathbb{E}[h(\mathbf{x}^{\text{m}})|\mathbf{x}^{\text{o}}] \approx \sum_{l=1}^{L} w_l h\left(\mathbf{x}^{\text{m}}_{(l)}\right)$, where $(\mathbf{x}^{\text{m}}_{(1)}, \mathbf{z}_{(1)}), \ldots, (\mathbf{x}^{\text{m}}_{(L)}, \mathbf{z}_{(L)})$ are i.i.d. samples from $p_{\boldsymbol{\theta}}(\mathbf{x}^{\text{m}}|\mathbf{x}^{\text{o}}, \mathbf{z}) q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x}^{\text{o}})$ that can be sampled via simple ancestral sampling, and, for all $l \in \{1, \ldots, L\}$

$$w_l = \frac{r_l}{r_1 + \ldots + r_L}, \text{ with } r_l = \frac{p_{\boldsymbol{\theta}}(\mathbf{x}^{\text{o}}|\mathbf{z}_{(l)}) p(\mathbf{z}_{(l)})}{q_{\boldsymbol{\gamma}}(\mathbf{z}_{(l)}|\mathbf{x}^{\text{o}})}. \tag{5}$$

Multiple Imputation with missIWAE. Multiple imputation is also approachable using similar computations. Indeed, using *sampling importance resampling* with the weights of Eq. (5) allows to draw samples that will be approximately i.i.d. samples from $p_{\boldsymbol{\theta}}(\mathbf{x}^{\text{m}}|\mathbf{x}^{\text{o}})$ when L is large.

3 Experiments

We wish to train a DLVM on an incomplete version of the static binarisation of MNIST (with 50% of the pixels missing uniformly at random), using the convolutional architecture of Salimans et al. (2015). To assess the validity of our claim that using a naive imputation function is not too harmful, we compare using the zero imputation function, and using an oracle imputation that utilises the true values of the missing pixels. The results are shown in Fig. 1. In the missVAE case ($K = 1$), using the oracle imputation provides a clear improvement. In the missIWAE case (with $K = 50$), both imputations schemes are on par (in accordance with our hypothesis), and outperform significantly the missVAE. As shown in Fig. 1, the missIWAE (with zero imputation) obtained is almost competitive

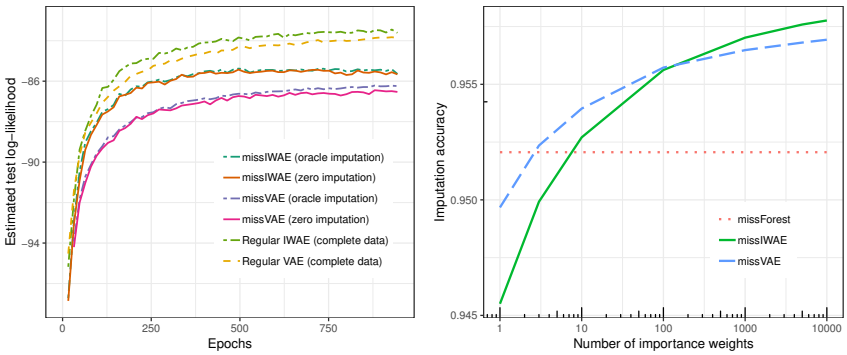

Figure 1: *Left:* Estimated test log-likelihood of various models trained on binary MNIST. Our missIWAE with zero imputation trained with half as much data as a regular VAE, and is less than 2 nats below. *Right :* Single imputation results for binary MNIST.

	Test accuracy	*Test cross-entropy*
Zero imputation	0.973880 (0.001832)	0.100334 (0.009197)
missForest imputation	0.980490 (0.001767)	0.064458 (0.006586)
missIWAE, mean imputation	0.984730 (0.000942)	0.051013 (0.003482)
missIWAE, multiple imputation	**0.986830 (0.000831)**	0.050900 (0.004439)
Complete data	0.986630 (0.000723)	**0.046384 (0.002628)**

Table 1: Test accuracy and cross-entropy obtained by training a convolutional network using the imputed versions of the static binarisation of MNIST. The numbers are the mean of 10 repeated trainings with different seeds and standard deviations are shown in brackets.

with a VAE trained on the complete MNIST data set. A few random samples from the missIWAE with zero imputation are displayed in Fig. 2.

We also evaluate the quality of the imputations provided by missIWAE. To this end, we use missVAE and missIWAE with zero imputation, together with the proposed importance sampling scheme for imputation, and compare them to a state-of-the-art single imputation algorithm: missForest (Stekhoven and Bühlmann, 2011). The results are displayed in Fig. 1. When $L \geq 10$, both missVAE and missIWAE outperform missForest, and missIWAE provides the most accurate imputations when $L \geq 1\,000$. Some simple imputation results are also presented in Fig. 3.

To evaluate multiple imputation, we consider the task of classifying the incomplete binary MNIST data set. We train a two-layer convolutional network (whose architecture is similar to the dropout one of Wan et al., 2013 and model selection is done using early stopping) using the original data and and some imputed versions, and assess the classification performance on the test set. Regarding missIWAE, we use both the single imputation obtained with $10\,000$ importance samples, and a multiple imputation of 20 complete data sets obtained using sampling importance resampling. Interestingly, *the convolutional network trained with the 20 imputed data sets outperforms the one trained on complete data* in terms of classification error (but not when it comes to the test cross-entropy). This suggests that the DLVM trained using missIWAE generalises quite well, and may also be used efficiently for data augmentation.

Figure 2: Random samples from the convolutional DLVM trained with the missIWAE bound. Half of the pixels of the MNIST training set were missing at random.

Figure 3: Random incomplete samples from the MNIST training data set, and the imputations obtained by missIWAE (trained with $K = 50$ importance weights, and imputed with $L = 10\,000$ importance weights).

References

S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Jozefowicz, and S. Bengio. Generating sentences from a continuous space. *Proceedings of CoNLL*, 2016.

Y. Burda, R. Grosse, and R. Salakhutdinov. Importance weighted autoencoders. In *Proceedings of the International Conference on Learning Representations*, 2016.

J. Domke and D. Sheldon. Importance weighting and varational inference. *Advances in Neural Information Processing Signals*, 2018.

M. Figurnov, S. Mohamed, and A. Mnih. Implicit reparameterization gradients. *Advances in Neural Information Processing Signals*, 2018.

R. Gómez-Bombarelli, J. N. Wei, D. Duvenaud, J. M. Hernández-Lobato, B. Sánchez-Lengeling, D. Sheberla, J. Aguilera-Iparraguirre, T. D. Hirzel, R. P. Adams, and A. Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, 2018.

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

P.-A. Mattei and J. Frellsen. Leveraging the exact likelihood of deep latent variable models. In *Advances in Neural Information Processing Systems*, 2018.

A. Nazabal, P. M. Olmos, Z. Ghahramani, and I. Valera. Handling incomplete heterogeneous data using VAEs. *arXiv preprint arXiv:1807.03653*, 2018.

D. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on Machine Learning*, pages 1278–1286, 2014.

C. P. Robert and G. Casella. *Monte Carlo statistical methods*. Springer-Verlag New York, 2004.

T. Salimans, D. P. Kingma, and M. Welling. Markov chain Monte Carlo and variational inference: Bridging the gap. In *International Conference on Machine Learning*, pages 1218–1226, 2015.

T. Salimans, A. Karpathy, X. Chen, and D. P. Kingma. Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications. *Proceedings of the International Conference on Learning Representations*, 2017.

D. J. Stekhoven and P. Bühlmann. MissForest—non-parametric missing value imputation for mixed-type data. *Bioinformatics*, 28(1):112–118, 2011.

L. Wan, M. Zeiler, S. Zhang, Y. LeCun, and R. Fergus. Regularization of neural networks using dropconnect. In *Proceedings of the 30th International Conference on Machine Learning*, pages 1058–1066, 2013.