Data Augmentation with Variational Autoencoders and Manifold Sampling

Clément Chadebec, Stéphanie Allassonnière

Data Augmentation with Variational Autoencoders and Manifold Sampling

Clément Chadebec and Stéphanie Allassonnière

Université de Paris, INRIA, Centre de recherche des Cordeliers, INSERM, Sorbonne Université, Paris, France
{clement.chadebec, stephanie.allassonniere}@inria.fr

Abstract. We propose a new efficient way to sample from a Variational Autoencoder in the challenging low sample size setting[1]. This method reveals particularly well suited to perform data augmentation in such a low data regime and is validated across various standard and *real-life* data sets. In particular, this scheme allows to greatly improve classification results on the OASIS database where balanced accuracy jumps from 80.7% for a classifier trained with the raw data to 88.6% when trained only with the synthetic data generated by our method. Such results were also observed on 3 standard data sets and with other classifiers.

Keywords: Data Augmentation · VAE · Latent space modelling

1 Introduction

Despite the apparent availability of always bigger data sets, the lack of data remains a key issue for many fields of application. One of them is medicine where practitioners have to deal with potentially very high dimensional data (*e.g.* functional Magnetic Resonance Imaging for neuroimaging) along with very low sample sizes (*e.g.* rare diseases or heterogeneous cancers) which make statistical analysis challenging and unreliable. In addition, the wide use of algorithms heavily relying on the deep learning framework [6] and requiring a large amount of data has made the need for data augmentation (DA) crucial to avoid poor performance or over-fitting [19]. As an example, a classic way to perform DA on images consists in applying simple transformations such as adding random noise, rotations etc. However, it may be easily understood that such augmentation techniques are strongly data dependent[2] and may still require the intervention of an expert assessing the relevance of the augmented samples. The recent development of generative models such as Generative Adversarial Networks (GAN) [7] or Variational AutoEncoders (VAE) [10,17] paves the way for consideration of another way to augment the training data. While GANs have already seen some success [4,22,2] and even for medical data [13,18] VAEs have been of least interest. One limitation of the use of both generative models relies in their need of a

[1] A code is available at https://github.com/clementchadebec/Data_Augmentation_with_VAE-DALI

[2] Think of digits where rotating a *6* gives a *9* for example.

large amount of data to be able to generate faithfully. In this paper, we argue that VAEs can actually be used to perform DA in challenging contexts provided that we amend the way we generate the data. Hence, we propose:

- A new non *prior-dependent* generation method using the learned geometry of the latent space and consisting in exploring it by sampling along geodesics.
- To use this method to perform DA in the small sample size setting on standard data sets and real data from OASIS database [16] where it allows to remarkably improve classification results.

2 Variational Autoencoder

Given a set of data $x \in \mathcal{X}$, a VAE aims at maximizing the likelihood of the associated parametric model $\{\mathbb{P}_\theta, \theta \in \Theta\}$. Assuming that there exist latent variables $z \in \mathcal{Z}$ living in a lower dimensional space $\mathcal{Z}$, the marginal distribution writes

$$p_\theta(x) = \int_{\mathcal{Z}} p_\theta(x|z)q(z)dz\,, \tag{1}$$

where q is a prior distribution over the latent variables and $p_\theta(x|z)$ is most of the time a simple distribution and is referred to as the *decoder*. A variational distribution q_φ (often taken as Gaussian) aiming at approximating the true posterior distribution and referred to as the *encoder* is then introduced. Using Importance Sampling allows to derive an unbiased estimate of $p_\theta(x)$ such that $\mathbb{E}_{z\sim q_\varphi}\big[\hat{p}_\theta\big] = p_\theta(x)$. Therefore, a lower bound on the logarithm of the objective function of Eq. (1) can be derived using Jensen's inequality:

$$\log p_\theta(x) \geq \mathbb{E}_{z\sim q_\varphi}\big[\log p_\theta(x,z) - \log q_\varphi(z|x)\big] = ELBO\,. \tag{2}$$

Using the reparametrization trick makes the ELBO tractable and so can be optimised with respect to both θ and φ, the *encoder* and *decoder* parameters. Once the model is trained, the decoder acts as a generative model and new data can be generated by simply drawing a sample using the prior q and feeding it to the decoder. Several axes of improvement of this model were recently explored. One of them consists in trying to bring geometry into the model by learning the latent structure of the data seen as a Riemannian manifold [3,5].

3 Some Elements on Riemannian Geometry

In the framework of differential geometry, one may define a Riemannian manifold $\mathcal{M}$ as a smooth manifold endowed with a Riemannian metric $\mathbf{G}$ which is a smooth inner product $\mathbf{G} : p \to \langle\cdot|\cdot\rangle_p$ on the tangent space $T_p\mathcal{M}$ defined at each point p of the manifold. The length of a curve γ between two points of the manifold $z_1, z_2 \in \mathcal{M}$ and parametrized by $t \in [0,1]$ such that $\gamma(0) = z_1$ and $\gamma(1) = z_2$ is given by $\mathcal{L}(\gamma) = \int_0^1 \|\dot{\gamma}(t)\|_{\gamma(t)}dt = \int_0^1 \sqrt{\langle\dot{\gamma}(t)|\dot{\gamma}(t)\rangle_{\gamma(t)}}dt$. Curves minimizing such

a length are called geodesics. For any $p \in \mathcal{M}$, the exponential map at p, Exp_p, maps a vector v of the tangent space $T_p\mathcal{M}$ to a point of the manifold $\widetilde{p} \in \mathcal{M}$ such that the geodesic starting at p with initial velocity v reaches $\widetilde{p}$ at time 1. In particular, if the manifold is *geodesically complete*, then Exp_p is defined on the entire tangent space $T_p\mathcal{M}$.

4 The Proposed Method

We propose a new sampling method exploiting the structure of the latent space seen as a Riemannian manifold and independent from the choice of the prior distribution. The view we adopt is to consider the VAE as a tool to perform dimensionality reduction by extracting the latent structure of the data within a lower dimensional space. Having learned such a structure, we propose to exploit it to enhance the data generation process. This differs from the fully probabilistic view which uses the prior to generate. We believe that this is far from being optimal since the prior appears quite strongly data dependent. We will adopt the same setting as [5] and so use a RHVAE since the metric used by the authors is easily computable, constraints geodesic path to travel through most populated areas of the latent space and the learned Riemannian manifold is geodesically complete. Nonetheless, the proposed method can be used with different metrics as well as long as the exponential map remains computable. We now assume that we are given a latent space with a Riemannian structure where the metric has been estimated from the input data.

4.1 The Wrapped Normal Distribution

The notion of normal distribution may be extended to Riemannian manifolds in several ways. One of them is the *wrapped* normal distribution. The main idea is to define a classic normal distribution $\mathcal{N}(0, \Sigma)$ on the tangent space $T_p\mathcal{M}$ for any $p \in \mathcal{M}$ and pushing it forward to the manifold using the exponential map. This defines a probability distribution on the manifold $\mathcal{N}^W(p, \Sigma)$ called the *wrapped* normal distribution. Sampling from this distribution is straight forward and consists in drawing a velocity in the tangent space from $\mathcal{N}(0, \Sigma)$ and mapping it onto the manifold using the exponential map [15]. Hence, the *wrapped* normal allows for a latent space prospecting along geodesic paths. Nonetheless, this requires to compute Exp_p which can be performed with a numerical scheme (see. App. C). On the left of Fig. 1 are displayed some geodesic paths with respect to the metric and different starting points (red dots) and initial velocities (orange arrows). Samples from $\mathcal{N}^W(p, I_d)$ are also presented in the middle and the right along with the encoded input data. As expected this distribution takes into account the local geometry of the manifold thanks to the geodesic shooting steps. This is a very interesting property since it encourages the samples to remain close to the data as geodesics tend to travel through locations with the lowest volume element $\sqrt{\det \mathbf{G}(z)}$ and so avoid areas with very poor information.

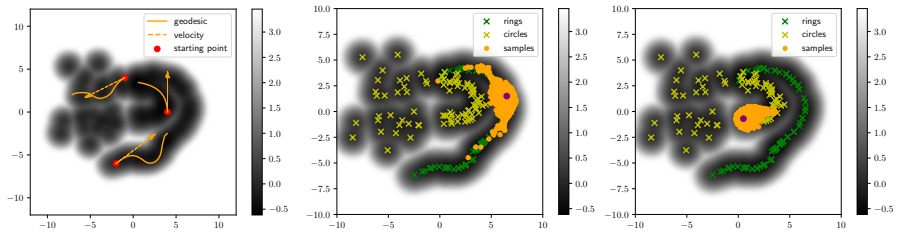

Fig. 1: *Left*: Geodesic *shooting* in a latent space learned by a RHVAE with different starting points (red dots) and initial velocities (orange arrows). *Middle and right*: Samples from the wrapped normal $\mathcal{N}^W(p, I_d)$. The log metric volume element $\log \sqrt{\det \mathbf{G}(z)}$ is presented in gray scale.

4.2 Riemannian Random Walk

A natural way to explore the latent space of a VAE consists in using a random walk like algorithm which moves from one location to another with a certain probability. The idea here is to create a *geometry-aware* Markov Chain $(z^t)_{t\in\mathbb{N}}$ where z^{t+1} is sampled using the *wrapped* normal $z^{t+1} \sim \mathcal{N}^W(z^t, \Sigma)$. However, a drawback of such a method is that every sample of the chain is accepted regardless of its relevance. Nonetheless, by design, the learned metric is such that it has a high volume element far from the data [5]. This implies that it encodes in a way the amount of information contained at a specific location of the latent space. The higher the volume element, the less information we have. The same idea was used in [11] where the author proposed to see the inverse metric volume element as a maximum likelihood objective to perform metric learning. In our case the likelihood definition writes

$$\mathcal{L}(z) = \frac{\rho_S(z)\sqrt{\det \mathbf{G}^{-1}(z)}}{\int_{\mathbb{R}^d} \rho_S(z)\sqrt{\det \mathbf{G}^{-1}(z)}dz}\,, \tag{3}$$

where $\rho_S(z) = 1$ if $z \in S$, 0 otherwise, and S is taken as a compact set so that the integral is well defined. Hence, we propose to use this measure to assess the samples quality as an *acceptance-rejection* rate α in the chain where $\alpha(\widetilde{z}, z) = \min\left(1, \frac{\sqrt{\det \mathbf{G}^{-1}(\widetilde{z})}}{\sqrt{\det \mathbf{G}^{-1}(z)}}\right)$, z is the current state of the chain and $\widetilde{z}$ is the proposal obtained by sampling from the *wrapped* Gaussian $\mathcal{N}^W(z, \Sigma)$. The idea is to compare the relevance of the proposed sample to the current one. The ratio is such that any new sample improving the likelihood metric $\mathcal{L}$ is automatically accepted while a sample degrading the measure is more likely to be rejected in the spirit of Hasting-Metropolis sampler. A pseudo-code is provided in Alg. 1.

Algorithm 1 Riemannian random walk

Input: z_0, Σ
for $t = 1 \rightarrow T$ **do**
  Draw $v_t \sim \mathcal{N}(0, \Sigma)$
  $\widetilde{z_t} \leftarrow \mathrm{Exp}_{z_{t-1}}(v_t)$
  Accept the proposal $\widetilde{z_t}$ with probability α
end for

4.3 Discussion

It may be easily understood that the choice of the covariance matrix Σ in Alg. 1 has quite an influence on the resulting sampling. On the one hand, a Σ with strong eigenvalues will imply drawing velocities of potentially high magnitude allowing for a better prospecting but proposals are more likely to be rejected. On the other hand, small eigenvalues involve a high acceptance rate but it will take longer to prospect the manifold. An adaptive method where Σ depends on $\mathbf{G}$ may be considered and will be part of future work.

Remark 1 *If Σ has small enough eigenvalues then Alg. 1 samples from Eq.* (3)

For the following DA experiments we will assume that Σ has small eigenvalues and so will sample directly using this distribution. See app. A for sampling results using the aforementioned method.

5 Data Augmentation Experiments For Classification

In this section, we explore the ability of the method to enrich data sets to improve classification results.

5.1 Augmentation Setting

We first test the augmentation method on three reduced data sets extracted from *well-known* databases MNIST and EMNIST. For MNIST, we select 500 samples applying either a balanced split or a random split ensuring that some classes are far more represented. For EMNIST, we select 500 samples from 10 classes such that they are composed of both lowercase and uppercase characters so that we end up with a small database with strong variability within classes. These data sets are then split such that 80% is allocated for training (referred to as the *raw data*) and 20% for validation. For a fair comparison, we use the original test set (*e.g.* ∼1000 samples per class for MNIST) to test the classifiers. This ensures statistically meaningful results while assessing the generalisation power on unseen data. We also validate the proposed DA method on the OASIS database which represents a nice example of day-to-day challenges practitioners have to face and is a benchmark database. We use 2D gray scale MR Images (208x176) with a mask notifying brain tissues and are referred to as the *masked*

Table 1: Summary of OASIS database demographics, mini-mental state examination (MMSE) and global clinical dementia rating (CDR) scores.

Data set	Label	Obs.	Age	Sex M/F	MMSE	CDR
OASIS	CN	316	45.1 ± 23.9	119/197	29.1 ± 1.1	0: 316
	AD	100	76.8 ± 7.1	41/59	24.3 ± 4.1	0.5: 70 , 1: 28, 2: 2
Train	CN	220	45.6 ± 23.6	86/134	29.1 ± 1.2	0: 220
	AD	70	77.4 ± 6.8	29/41	23.7 ± 4.3	0.5: 47 , 1: 21, 2: 2
Val	CN	30	48.9 ± 24.1	11/19	29.2 ± 0.8	0: 30
	AD	12	75.4 ± 7.2	4/8	25.8 ± 4.2	0.5: 7 , 1: 5, 2: 0
Test	CN	66	41.7 ± 24.3	22/44	29.0 ± 1.0	0: 66
	AD	18	75.1 ± 7.5	8/10	25.8 ± 2.7	0.5: 16 , 1: 2, 2: 0

T88 images in [16]. We refer the reader to their paper for further image preprocessing details. We consider the binary classification problem consisting in trying to detect MRI of patients having been diagnosed with Alzheimer Disease (AD). We split the 416 images into a training set (70%) (*raw data*), a validation set (10%) and a test set (20%). A summary of demographics, mini-mental state examination (MMSE) and global clinical dementia rating (CDR) is made available in Table. 1. On the one hand, for each data set, the train set (*raw data*) is augmented by a factor 5, 10 and 15 using classic DA methods (random noise, cropping etc.). On the other hand, VAE models are trained individually on each class of the *raw data*. The generative models are then used to produce 200, 500, 1k or 2k synthetic samples per class with either the classic generation scheme (*i.e.* the prior) or the proposed method. We then train classifiers with 5 independent runs on 1) the *raw data*; 2) the augmented data using basic transformations; 3) the augmented data using the VAE models; 4) only the synthetic data generated by the VAEs. A DenseNet model[3] [8] is used for the toy data while we also train hand made MLP and CNN models on OASIS (See App. E). The main metrics obtained on the test set are reported in Tables. 2 and 3.

5.2 Results

Toy Data As expected generating new samples using the proposed method improves their relevance. The method indeed allows for a quite impressive gain in the model accuracy when synthetic samples are added to the real ones (leftmost column of Table. 2). This is even more striking when looking at the rightmost column where only synthetic samples are used to train the classifier. For instance, when only 200 synthetic samples per class for MNIST are generated with a VAE and used to train the classifier, the classic method fails to produce meaningful samples since a loss of 20 pts in accuracy is observed when compared to the *raw data*. Interestingly, our method seems to avoid such an effect. Even more

[3] We use the code in [1] (See App. E).

Table 2: DA on *toy* data sets. Mean accuracy and standard deviation across 5 independent runs are reported. In gray are the cells where the accuracy is higher on synthetic data than on the *raw data*.

Data sets	MNIST	MNIST**	EMNIST**	MNIST	MNIST**	EMNIST**
Raw data	89.9 (0.6)	81.6 (0.7)	82.6 (1.4)	-	-	-
	Raw + Synthetic			Synthetic only		
Aug. (X5)	92.8 (0.4)	86.5 (0.9)	85.6 (1.3)	-	-	-
Aug. (X10)	88.3 (2.2)	82.0 (2.4)	85.8 (0.3)	-	-	-
Aug. (X15)	92.8 (0.7)	85.9 (3.4)	86.6 (0.8)	-	-	-
VAE-200*	88.5 (0.9)	84.1 (2.0)	81.7 (3.0)	69.9 (1.5)	64.6 (1.8)	65.7 (2.6)
VAE-500*	90.4 (1.4)	87.3 (1.2)	83.4 (1.6)	72.3 (4.2)	69.4 (4.1)	67.3 (2.4)
VAE-1k*	91.2 (1.0)	86.0 (2.5)	84.4 (1.6)	83.4 (2.4)	74.7 (3.2)	75.3 (1.4)
VAE-2k*	92.2 (1.6)	88.0 (2.2)	86.0 (0.2)	86.6 (2.2)	79.6 (3.8)	78.9 (3.0)
rhvae-200*	89.9 (0.5)	82.3 (0.9)	83.0 (1.3)	76.0 (1.8)	61.5 (2.9)	59.8 (2.6)
rhvae-500*	90.9 (1.1)	84.0 (3.2)	84.4 (1.2)	80.0 (2.2)	66.8 (3.3)	67.0 (4.0)
rhvae-1k*	91.7 (0.8)	84.7 (1.8)	84.7 (2.4)	82.0 (2.9)	69.3 (1.8)	73.7 (4.1)
rhvae-2k*	92.7 (1.4)	86.8 (1.0)	84.9 (2.1)	85.2 (3.9)	77.3 (3.2)	68.6 (2.3)
Ours-200*	91.0 (1.1)	84.1 (2.0)	85.1 (1.1)	87.2 (1.1)	79.5 (1.6)	77.1 (1.6)
Ours-500*	92.3 (1.1)	87.7 (0.9)	85.1 (1.1)	89.1 (1.3)	80.4 (2.1)	80.2 (2.0)
Ours-1k*	93.3 (0.8)	**89.7 (0.8)**	87.0 (1.0)	90.2 (1.4)	86.2 (1.8)	82.6 (1.3)
Ours-2k*	**94.3 (0.8)**	89.1 (1.9)	**87.6 (0.8)**	**92.6 (1.1)**	**87.6 (1.3)**	**86.0 (1.0)**

* Number of generated samples ** Unbalanced data sets

impressive is the fact that we are able to produce synthetic data sets on which the classifier outperforms greatly the results observed on the *raw data* (3 to 6 pts gain in accuracy) while keeping a relatively low standard deviation (see gray cells). Secondly, this example also shows why geometric DA is still questionable and remains data dependent. For instance, augmenting the *raw data* by a factor 10 (including flips and rotations) does not seem to have a notable effect on the MNIST data sets but still improves results on EMNIST. On the contrary, our method seems quite **robust to data set changes**.

OASIS Balanced accuracy obtained on OASIS with 3 classifiers is made available in Table. 3. In this experiment, using the new generation scheme again improves overall the metric for each classifier when compared to the *raw data* and other augmentation methods. Moreover, the strong relevance of the created samples is again supported by the fact that the classifiers are again able to strongly outperform the results on the *raw data* even when trained only with synthetic ones. Finally, the method appears **robust to classifiers** and can be used with high-dimensional complex data such as MRI.

6 Conclusion

In this paper, we proposed a new way to generate new data from a Variational Autoencoder which has learned the latent geometry of the input data. This method was then used to perform DA to improve classification tasks in the low

Table 3: DA on OASIS data base. Mean balanced accuracy on independent 5 runs with several classifiers.

Networks	MLP		CNN		Densenet	
Raw data	80.7 (4.1)	-	72.5 (3.5)	-	77.4 (3.3)	-
	Raw + Synthetic	Synthetic Only	Raw + Synthetic	Synthetic Only	Raw + Synthetic	Synthetic Only
Aug. (X5)	84.3 (1.3)	-	80.0 (3.5)	-	73.9 (5.1)	-
Aug. (X10)	76.0 (2.8)	-	82.8 (3.7)	-	78.3 (4.1)	-
Aug. (X15)	78.7 (5.3)	-	80.3 (3.7)	-	76.6 (1.1)	-
VAE-200*	80.7 (1.5)	77.8 (1.3)	79.4 (3.6)	65.0 (12.3)	76.5 (3.2)	74.0 (3.0)
VAE-500*	79.7 (1.4)	77.4 (1.5)	72.6 (7.0)	70.2 (5.0)	74.9 (4.3)	72.8 (1.8)
VAE-1000*	81.3 (0.0)	76.5 (0.6)	74.4 (9.4)	73.0 (3.3)	73.5 (1.3)	74.9 (2.6)
VAE-2000*	80.7 (0.3)	78.1 (1.6)	71.1 (4.9)	76.9 (2.6)	74.0 (4.9)	73.3 (3.4)
Ours-200*	84.3 (0.0)	86.7 (0.4)	76.4 (5.0)	75.4 (6.6)	78.2 (3.0)	74.3 (4.8)
Ours-500*	**87.2 (1.2)**	**88.6 (1.1)**	81.8 (4.6)	81.8 (3.7)	80.2 (2.8)	**84.2 (2.8)**
Ours-1000*	84.2 (0.3)	84.4 (1.8)	83.5 (3.2)	79.8 (2.8)	82.2 (4.7)	76.7 (3.8)
Ours-2000*	85.3 (1.9)	84.2 (3.3)	**84.5 (1.9)**	**83.9 (1.9)**	**82.9 (1.8)**	73.6 (5.8)

* Number of generated samples

sample size setting on both toy and real data and with different kind of classifiers. In each case, the method allows for an impressive gain in the classification metrics (*e.g.* balanced accuracy jumps from 80.7 to 88.6 on OASIS). Moreover, the relevance of the generated data was supported by the fact that classifiers were able to perform better when trained with only synthetic data than on the *raw data* in all cases. Future work would consist in using the method on even more challenging data such as 3D volumes and using smaller data sets.

Acknowledgment

The research leading to these results has received funding from the French government under management of Agence Nationale de la Recherche as part of the "Investissements d'avenir" program, reference ANR-19-P3IA-0001 (PRAIRIE 3IA Institute) and reference ANR-10-IAIHU-06 (Agence Nationale de la Recherche-10-IA Institut Hospitalo-Universitaire-6). Data were provided in part by OASIS: Cross-Sectional: Principal Investigators: D. Marcus, R, Buckner, J, Csernansky J. Morris; P50 AG05681, P01 AG03991, P01 AG026276, R01 AG021910, P20 MH071616, U24 RR021382

References

1. Amos, B.: bamos/densenet.pytorch (2020), https://github.com/bamos/densenet.pytorch, original-date: 2017-02-09T15:33:23Z
2. Antoniou, A., Storkey, A., Edwards, H.: Data augmentation generative adversarial networks. arXiv:1711.04340 [cs, stat] (2018)
3. Arvanitidis, G., Hansen, L.K., Hauberg, S.: Latent space oddity: On the curvature of deep generative models. In: 6th International Conference on Learning Representations, ICLR 2018 (2018)
4. Calimeri, F., Marzullo, A., Stamile, C., Terracina, G.: Biomedical data augmentation using generative adversarial neural networks. In: Lintas, A., Rovetta, S., Verschure, P.F., Villa, A.E. (eds.) Artificial Neural Networks and Machine Learning – ICANN 2017, vol. 10614, pp. 626–634. Springer International Publishing (2017), http://link.springer.com/10.1007/978-3-319-68612-7_71, series Title: Lecture Notes in Computer Science
5. Chadebec, C., Mantoux, C., Allassonnière, S.: Geometry-aware hamiltonian variational auto-encoder. arXiv:2010.11518 [cs, math, stat] (2020)
6. Goodfellow, I., Bengio, Y., Courville, A., Bengio, Y.: Deep learning, vol. 1. MIT press Cambridge (2016), issue: 2
7. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems. pp. 2672–2680 (2014)
8. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2261–2269. IEEE (2017)
9. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
10. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv:1312.6114 [cs, stat] (2014)
11. Lebanon, G.: Metric learning for text documents. IEEE Transactions on Pattern Analysis and Machine Intelligence **28**(4), 497–508 (2006)
12. LeCun, Y.: The MNIST database of handwritten digits (1998)
13. Liu, Y., Zhou, Y., Liu, X., Dong, F., Wang, C., Wang, Z.: Wasserstein gan-based small-sample augmentation for new-generation artificial intelligence: a case study of cancer-staging data in biology. Engineering **5**(1), 156–163 (2019)
14. Louis, M.: Computational and statistical methods for trajectory analysis in a Riemannian geometry setting. PhD Thesis, Sorbonnes universités (2019)
15. Mallasto, A., Feragen, A.: Wrapped gaussian process regression on riemannian manifolds. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5580–5588. IEEE (2018)
16. Marcus, D.S., Wang, T.H., Parker, J., Csernansky, J.G., Morris, J.C., Buckner, R.L.: Open access series of imaging studies (OASIS): Cross-sectional MRI data in young, middle aged, nondemented, and demented older adults. Journal of Cognitive Neuroscience **19**(9), 1498–1507 (2007)
17. Rezende, D.J., Mohamed, S., Wierstra, D.: Stochastic backpropagation and approximate inference in deep generative models. In: International conference on machine learning. pp. 1278–1286. PMLR (2014)
18. Sandfort, V., Yan, K., Pickhardt, P.J., Summers, R.M.: Data augmentation using generative adversarial networks (CycleGAN) to improve generalizability in CT segmentation tasks. Scientific reports **9**(1), 16884 (2019)

19. Shorten, C., Khoshgoftaar, T.M.: A survey on Image Data Augmentation for Deep Learning. Journal of Big Data **6**(1), 60 (2019)
20. Tomczak, J., Welling, M.: Vae with a vampprior. In: International Conference on Artificial Intelligence and Statistics. pp. 1214–1223. PMLR (2018)
21. Xiao, H., Rasul, K., Vollgraf, R.: Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747 (2017)
22. Zhu, X., Liu, Y., Qin, Z., Li, J.: Data augmentation in emotion classification using generative adversarial networks. arXiv:1711.00648 [cs] (2017)

Appendix A: Comparison with Prior-Based Methods

In this section, we compare the samples quality between *prior-based* methods and ours on various standard and *real-life* data sets.

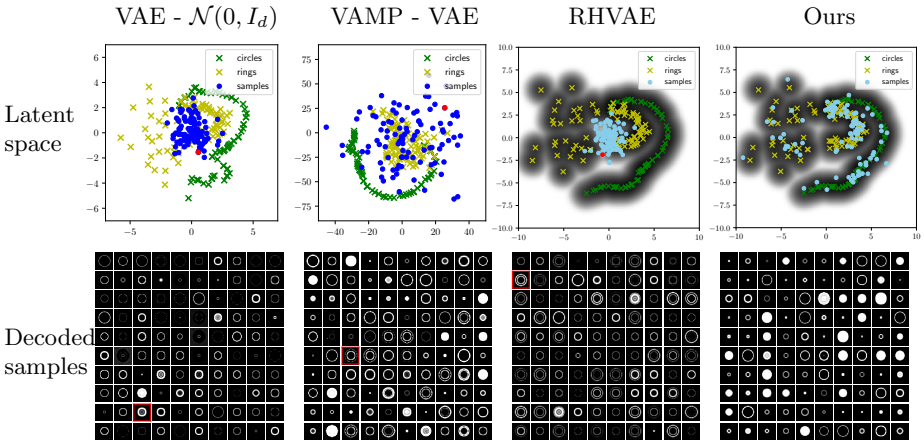

Fig. 2: Comparison between prior-based generation methods and the proposed Riemannian random walk (ours). Top: the learned latent space with the encoded training data (crosses) and 100 samples for each method (blue dots). Bottom: the resulting decoded images. The models are trained on 180 binary circles and rings with the same neural network architectures.

Standard Data Sets The method is first validated on a hand-made synthetic data set composed of 180 binary images of circles and rings of different diameters and thicknesses (see training samples in Fig. 3). We then train a VAE with a normal prior, a VAE with a VAMP prior [20] and a RHVAE until the ELBO does not improve for 50 epochs. Any relevant parameters setting is stated in App. D.

Fig. 2 highlights the obtained samplings with each model using either the *prior-based* generation procedure or the one proposed in this paper. The first row presents the learned latent space along with the means of the posteriors associated to the training data (crosses) and 100 latent space samples for each generation method (blue dots). The second row displays the corresponding decoded images. The first outcome of such a study is that sampling from the prior distribution $\mathcal{N}(0, I_d)$ leads to a poor latent space prospecting. Therefore, even with balanced training classes, we end up with a model over-representing certain elements of a given class (rings). This is even more striking with the RHVAE since it tends to stretch the resulting latent space. This effect seems nonetheless

mitigated by the use of a multimodal prior such as the VAMP. However, another limitation of *prior-based* methods is that they may sample in locations of the latent space potentially containing very few information (*i.e.* where no data is available). Since the decoder appears to interpolate quite linearly, the classic scheme will generate images which mainly correspond to a superposition of samples (see an example with the red dots in Fig. 2 and the corresponding samples framed in red). Moreover, there is no way to assess a sample quality before decoding it and assessing visually its relevance. These limitations may lead to a (very) poor representation of the actual data set diversity while presenting quite a few *irrelevant* samples. Impressively, sampling along geodesic paths leads to far more diverse and sharper samples. The new sampling scheme avoids regions that have been poorly prospected so that almost every decoded sample is visually satisfying and accounts for the data set diversity. In Fig. 3, we also compare the models on a *reduced* MNIST [12] data set composed of 120 samples of 3 different classes and a *reduced* FashionMNIST [21] data set composed again of 120 samples from 3 distinct classes. The models are trained with the same neural network architectures, batch size and learning rate. An early stopping strategy is adopted and consists in stopping training if the ELBO does not improve for 50 epochs. As discussed earlier, changing the prior may indeed improve the model generation capacity. For instance samples from the VAE with the VAMP prior (3^{rd} row of Fig. 3) are closer to the training data (1^{st} row of Fig. 3) than with the Gaussian prior (2^{nd} and 4^{th} row). The model is for instance able to generate circles when trained with the synthetic data while models using a standard normal prior are not. Nonetheless, a non negligible part of the generated samples are degraded (see saturated images for the *reduced* MNIST data for instance). This aspect is mitigated with the proposed generation method which generates more diverse and sharper samples.

OASIS Database The new generation scheme is then assessed on the publicly available OASIS database composed of 416 patients aged 18 to 96, 100 of whom have been diagnosed with very mild to moderate Alzheimer disease (AD). A VAE and a RHVAE are then trained to generate either cognitively normal (CN) or AD patients with the same early stopping criteria as before. Fig. 4 shows samples extracted from the training set (top), MRI generated by a vanilla VAE (2^{nd} row) and images from the Riemannian random walk we propose (3^{rd} row). For each method, the upper row shows images of patients diagnosed CN while the bottom row presents an AD diagnosis. Again the proposed sampling seems able to generate a wider range of sharp samples while the VAE appears to produce non-realistic degraded images which are very similar (see red frames). For example, the proposed scheme allows us to generate realistic *old*[4] patients with no AD (blue frames) or younger patients with AD (orange frames) even though they are under-represented in the training set. Generating 100 images of

[4] An older person is characterised by larger ventricles.

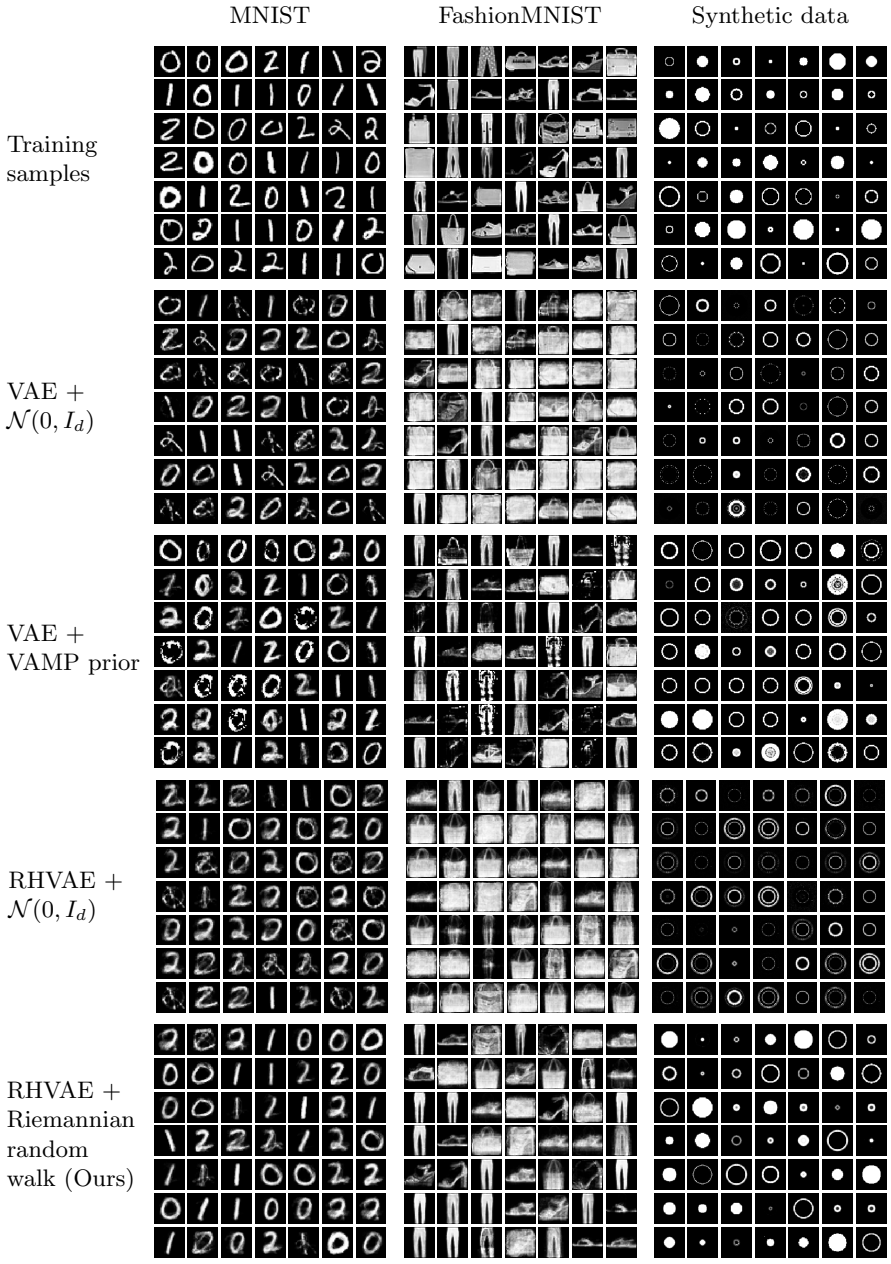

Fig. 3: Comparison of 4 sampling methods on the *reduced* MNIST, *reduced* Fashion and the synthetic data sets. From top to bottom: 1) samples extracted from the training set; 2) samples generated with a Vanilla VAE and using the prior; 3) from the VAMP prior VAE ; 4) from a RHVAE and the *prior-based* generation scheme; 5) from a RHVAE and using the proposed Riemannian random walk. All the models are trained with the same encoder and decoder networks and identical latent space dimension. An early stopping strategy is adopted and consists in stopping training if the ELBO does not improve for 50 epochs.

OASIS database takes 1 min. with the proposed method and 40 sec.[5] with Intel Core i7 CPU (6x1.1GHz) and 16 GB RAM.

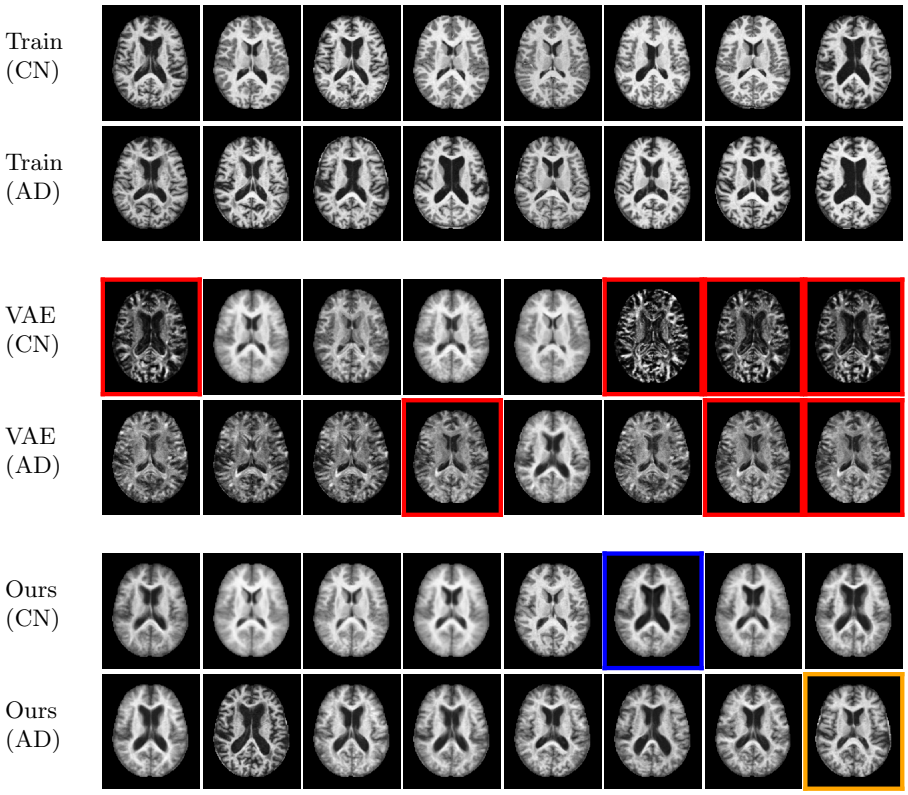

Fig. 4: Generation of CN or AD patients from the OASIS database. Training samples (top), generation with a VAE and normal prior (2^{nd} row) and with the Riemannian random walk (bottom). Generating using the prior leads to either unrealistic images or similar samples (red frames) while the proposed method generates sharper and more diverse samples. For instance, it is able to generate CN *older* patients (blue frames) or younger AD (orange frames) even though they are under-represented within the training set.

[5] Depends on the chains' length (here 200 steps per image).

Appendix B: Discussion of Remark. 1

Remark 2 *If Σ has small enough eigenvalues then Alg. 1 samples from*

$$\mathcal{L}(z) = \frac{\rho_S(z)\sqrt{\det \mathbf{G}^{-1}(z)}}{\int\limits_{\mathbb{R}^d} \rho_S(z)\sqrt{\det \mathbf{G}^{-1}(z)}dz}\,, \tag{4}$$

where $\rho_S(z) = 1$ if $z \in S$, 0 otherwise, and S is taken as a compact set so that the integral is well defined.

If Σ has small enough eigenvalues, it means that the initial velocity $v \sim \mathcal{N}(0, \Sigma)$ will have a low magnitude with high probability. In such a case, we can show with some approximation that the ratio α in the Riemannian random walk is a Hasting-Metropolis ratio with target density given by Eq. (4). We recall that the classic Hasting-Metropolis ratio writes

$$\alpha(x, y) = \frac{\pi(y)}{\pi(x)} \cdot \frac{q(x, y)}{q(y, x)}\,,$$

where π is the target distribution and q a proposal distribution. In the case of small magnitude velocities, one may show that q is symmetric that is

$$q(x, y) = q(y, x)\,.$$

In our setting, a proposal $\widetilde{z}$ is made by computing the geodesic γ starting at $\gamma(0) = z$ with initial velocity $\dot{\gamma}(0) = v$ where $v \sim \mathcal{N}(0, \Sigma)$ and evaluating it at time 1. First, we remark that γ is well defined since the Riemnanian manifold $\mathcal{M} = (\mathbb{R}, g)$ is *geodesically complete* and γ is unique. Moreover, we have that $\overleftarrow{\gamma}$ is the unique geodesic with initial position $\overleftarrow{\gamma}(0) = \widetilde{z} = \gamma(1)$ and initial velocity $\dot{\overleftarrow{\gamma}}(0) = -\dot{\gamma}(1)$ and we have

$$\overleftarrow{\gamma}(t) = \gamma(1 - t), \qquad \forall\, t \in [0, 1]\,.$$

In the case of small enough initial velocity, a taylor expansion of the exponential may be performed next to $0 \in T_z\mathcal{M}$ and consists in approximating geodesic curves with straight lines. That is, for $t \in [0, 1]$

$$\mathrm{Exp}_z(vt) \approx z + vt\,,$$

where $v = \widetilde{z} - z$. In such a case we have

$$\dot{\gamma}(t) = v = \dot{\gamma}(0) = \dot{\gamma}(1) = -\dot{\overleftarrow{\gamma}}(0)\,.$$

Moreover, we have on the one hand

$$q(\widetilde{z}, z) = p(\widetilde{z}|z) = p(\mathrm{Exp}_z(v)|z) \simeq \mathcal{N}(z, \Sigma)\,.$$

On the other hand

$$q(z, \widetilde{z}) = p(z|\widetilde{z}) = p(\mathrm{Exp}_{\widetilde{z}}(-v)|\widetilde{z}) \simeq \mathcal{N}(\widetilde{z}, \Sigma)$$

Therefore,

$$q(\widetilde{z}, z) = q(z, \widetilde{z})\,.$$

Finally, the ratio α in the Riemannian random walk may be seen as a Hasting-Metropolis ratio where the target density is given by Eq. (3) and so the algorithm samples from such a distribution.

Appendix C: Computing the Exponential map

To compute the exponential map at any given point $p \in \mathcal{M}$ and for any tangent vector $v \in T_p\mathcal{M}$ we rely on the Hamiltonian definition of geodesic curves. First, for any given $v \in T_p\mathcal{M}$, the linear form:

$$q_v : \begin{cases} T_p\mathcal{M} \to \mathbb{R} \\ u \quad \to g_p(u, v) \end{cases},$$

is called a moment and is a representation of v in the dual space. In short, we may write $q_v(u) = u^\top \mathbf{G} v$. Then, the definition of the Hamiltonian follows

$$H(p, q) = \frac{1}{2} g_p^*(q, q)\,,$$

where g_p^* is the dual metric whose local representation is given by $\mathbf{G}^{-1}(p)$, the inverse of the metric tensor. Finally, all along geodesic curves the following equations hold

$$\frac{\partial p}{\partial t} = \frac{\partial H}{\partial q}, \qquad \frac{\partial q}{\partial t} = -\frac{\partial H}{\partial p}\,. \tag{5}$$

Such a system of differential equations may be integrated pretty straight forwardly using simple numerical schemes such as the second order Runge Kutta integration method and Alg. 2 as in [14]. Noteworthy is the fact that such an algorithm only involves one metric tensor inversion at initialization to recover the initial moment from the initial velocity. Moreover, it involves closed form operations since the inverse metric tensor $\mathbf{G}^{-1}$ is known (in our case) and so the gradients in Eq. (5) can be easily computed.

Algorithm 2 Computing the Exponential map

Input: $z_0 \in \mathcal{M}$, $v \in T_{z_0}\mathcal{M}$ and T
$q \leftarrow \mathbf{G} \cdot v$
$dt \leftarrow \frac{1}{T}$
for $t = 1 \to T$ **do**
  $p_{t+\frac{1}{2}} \leftarrow p_t + \frac{1}{2} \cdot dt \cdot \nabla_q H(p_t, q_t)$
  $q_{t+\frac{1}{2}} \leftarrow q_t - \frac{1}{2} \cdot dt \cdot \nabla_p H(p_t, q_t)$
  $p_{t+1} \leftarrow p_t + dt \cdot \nabla_q H(p_{t+\frac{1}{2}}, q_{t+\frac{1}{2}})$
  $q_{t+1} \leftarrow q_t - dt \cdot \nabla_p H(p_{t+\frac{1}{2}}, q_{t+\frac{1}{2}})$
end for
Return p_T

Appendix D: VAEs Parameters Setting

Table. 4 summarizes the main hyper-parameters we use to perform the experiments presented in the paper while Table. 5 shows the neural networks architectures employed. As to training parameters, we use a Adam optimizer [9] with a learning of 10^{-3}. For the augmentation experiments, we stop training if the ELBO does not improve for 20 epochs for all data sets except for OASIS where the learning rate is decreased to 10^{-4} and training is stopped if the ELBO does not improve for 50 epochs.

Table 4: RHVAE parameters for each data set.

Data sets	Parameters					
	d^*	n_{lf}	ε_{lf}	T	λ	$\sqrt{\beta_0}$
Synthetic	2	3	10^{-2}	0.8	10^{-3}	0.3
reduced Fashion	2	3	10^{-2}	0.8	10^{-3}	0.3
MNIST (bal.)	2	3	10^{-2}	0.8	10^{-3}	0.3
MNIST (unbal.)	2	3	10^{-2}	0.8	10^{-3}	0.3
EMNIST	2	3	10^{-2}	0.8	10^{-3}	0.3
OASIS	2	3	10^{-3}	0.8	10^{-2}	0.3

* Latent space dimension (same for VAE and VAMP-VAE)

Table 5: Neural networks architectures of the VAE, VAMP-VAE and RHVAE for each data set. The *encoder* and *decoder* are the same for all models.

Synthetic, MNIST & Fashion			
Net	Layer 1	Layer 2	Layer 3
μ_ϕ^*	(D, 400, relu)	(400, d, lin.)	-
Σ_ϕ^*		(400, d, lin.)	-
π_θ^*	(d, 400, relu)	(400, D, sig.)	-
L_ψ (diag.)	(D, 400, relu)	(400, d, lin.)	-
L_ψ (low.)		(400, $\frac{d(d-1)}{2}$, lin.)	-
OASIS			
μ_ϕ^*	(D, 1k, relu)	(1k, 400, relu)	(400, d, lin.)
Σ_ϕ^*			(400, d, lin.)
π_θ^*	(d, 400, relu)	(400, 1k, relu)	(1k, D, sig.)
L_ψ (diag.)	(D, 400, relu)	(400, d, lin.)	-
L_ψ (low.)		(400, $\frac{d(d-1)}{2}$, lin.)	-

* Same for all VAE models

Appendix E: Classifier Parameter Setting

As to the models used as benchmark for data augmentation, the DenseNet implementation we use is the one in [1] with a *growth rate* equals to 10, *depth* of 20 and 0.5 *reduction* and is trained with a learning rate of 10^{-3}. For OASIS, the MLP has 400 hidden units and relu activation function and the CNN is as follows

Table 6: CNN classifier architecture used. Each convolutional block has a padding of 1.

Layer	Architectures
input	(1, 208, 176)
Layer 1	Conv2D(1, 8, kernel=(3, 3), stride=1) Batch normalization LeakyRelu Maxpool (2, 2, stride=2)
Layer 2	Conv2D(8, 16, kernel=(3, 3), stride=1) Batch normalization LeakyRelu Maxpool (2, 2, stride=2)
Layer 3	Conv2D(16, 32, kernel=(3, 3), stride=2) Batch normalization LeakyRelu Maxpool (2, 2, stride=2)
Layer 4	Conv2D(32, 64, kernel=(3, 3), stride=2) Batch normalization LeakyRelu Maxpool (2, 2, stride=2)
Layer 5	MLP(256, 100) Relu
Layer 6	MLP(100, 2) Log Softmax

For the toy data, the DenseNet is trained until the loss does not improve on the validation set for 50 epochs. On OASIS, we make a random search on the learning rate for each model chosen in the range $[10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}]$. The model is trained on 5 independent runs with the same learning rate and keep the model achieving the best mean balanced accuracy on the validation set. For the CNN and MLP, we stop training if the validation loss does not improve for 20 epochs and for the densenet training is stopped if no improvement is observed on the validation loss for 10 epochs. Each model is trained with an Adam optimizer.