Divergence-free continuous normalizing flows for uncertainty quantification

Simon Benaïchouche, Guillaume Morel, François Rousseau, Ronan Fablet

Divergence-free continuous normalizing flows for uncertainty quantification

Simon Benaïchouche,[1,3] **Guillaume Morel,** [2] **François Rousseau,** [2] **Ronan Fablet** [1]

[1] IMT Atlantique, UMR CNRS Lab-STICC, Brest, FR
[2] IMT Atlantique, LaTIM U1101 INSERM, Brest, FR
[3] Eodyn, Brest, FR
simon.benaichouche@imt-atlantique.fr, guillaume.morel@imt-atlantique.fr, francois.rousseau@imt-atlantique.fr, ronan.fablet@imt-atlantique.fr

Abstract

Uncertainty quantification in ill-posed inverse problems is a critical issue in a variety of scientific domains, including among others signal processing, imaging science, geoscience, remote sensing.... This has led to a variety of approaches, especially using Bayesian schemes such as Kalman methods, particle filtering schemes and variational Bayesian inference. Dealing with non-linear and non-Gaussian processes remain however a challenge, especially when considering high-dimensional systems. Recently, normalizing flows using deep neural networks have emerged as a very powerful tool to train generative models, which can sample realistic states, while making feasible the computation of the likelihood function. Their exploitation for uncertainty quantification in terms of differential Shannon entropy however requires the use Monte Carlo methods whose computational cost can be prohibitive, especially for high-dimensional systems such as time, space and space-time processes. Here, we introduce a new class of continuous normalizing flows with a divergence-free constraint for the underlying governing ordinary differential equations. This divergence-free constraint results in the preservation of the differential Shannon entropy through the trained flows.

We demonstrate the relevance of the proposed framework to reach state-of-the-art performance for generative modeling tasks. We also illustrate applications to uncertainty quantification for the reconstruction of 1D and 2D states from partial observations. We discuss further our main contributions and applications to real-world case-studies.

Introduction

The reconstruction of hidden states from partial and/or noisy observations, referred to as inverse problems, are key challenges in numerous domains such as signal processing, computational imaging, medical imaging, remote sensing, geoscience,..... Model-driven schemes have long been the state-of-the-art methods, especially when dealing with physical states (Evensen 2009). They state the inverse problem as the minimization problem with two terms, a data-fidelity or observation term J and a re regularization term R:

$$\hat{X} = \arg\min_X J(X, Y) + R(Y) \tag{1}$$

with Y the observation and X the state to be reconstructed. The observation terms involves the definition of the forward observation model to relate X and Y. Regularization term states some prior on considered problem. For example when dealing with physics-driven dynamical systems, prior R is stated as $R(.) := \sum_{k=1}^{T} \|\mathcal{M}(x_{k-1}) - x_k\|$ where $\mathcal{M}$ denotes a forecast operator associated to a physical equations (Courtier and Talagrand 1987). Recently, data-driven and deep learning methods have arised as relevant alternatives to address inverse problems. While deep learning approaches lead to the state-of-art-performance for inverse problems (Dong et al. 2014), (Minaee et al. 2021) (Xie, Xu, and Chen 2012) (Yu et al. 2018), they also compete with model-driven schemes for the reconstruction and forecasting of geophysical systems (Fablet et al. 2021; Nonnenmacher and Greenberg 2021).

These method are generally trained by gradient descent to minimize a given reconstruction cost.

$$\theta^* = \arg\min E(d(\mu_\theta(Y), X)) \tag{2}$$

it is known (Bishop 2006) that when 2 is treated as a regression problem with least square estimation, a minimizer of 2 is given by the expectation of the conditional density $p(X|Y)$. The estimation of $p(X|Y)$ is of a key importance for in order to transpose many applications of a Data assimilation task in a stochastic perspective : One would like to take decision on the basis of a risk minimization for a wide range of applications instead of taking a decision relying only on the estimated mean μ_θ. Furthermore, the knowledge of the probability density $p(.|Y)$ allows the computation of information theoretic related quantities such as the differential entropy : this quantities extends the classical Shannon entropy (Shannon and Weaver 1949) to continuous variable. For a random variable X with associated density function p, the differential entropy of X is defined by:

$$H(X) := E_p[-log(p(x))] = -\int_\Omega p(x) log(p(x)).dx \tag{3}$$

It can be understood as a measure of uncertainty : a random variable with high concentration of the measure would trend to have low entropy. One can give two examples for 1-d variables : dirac distributions δ (no uncertainty) for which $H(\delta) = -\infty$ or uniform distribution U on $\Omega = [a, b]$, which corresponds to a case of maximal uncertainty verifies $H(U) = ln(b - a) \to +\infty$ if $|b - a| \to +\infty$. and is of a key importance for applications such as sensor placement (Yin et al. 2017), However for an arbitrary distribution

f, the computation of the differential Shannon entropy relies on Monte Carlo methods (James 1980) whose computational cost may be prohibitive for real-world processes.Thus derivation of approximation method of the differential entropy is an active area of researches (Huber et al. 2008; Hyvärinen 1997; Ao and Li 2022).

Here, we address uncertainty quantification according to the differential Shannon entropy in inverse problems, while keeping analytical an computation of the differential Shannon entropy to overcome the shortcomings of Monte-Carlo schemes for high-dimensional systems. We state the proposed framework as the learning constrained continuous normalizing flows (Chen et al. 2018). Our main contributions are as follows:

- We introduce a novel class of divergence-free continuous normalizing flows. We show that the divergence-free constraint guarantees the preservation of information-theoretic quantities such as the differential Shannon entropy.
- We present a neural implementation of the proposed scheme with an explicit divergence-preserving parameterization of the neural architecture.
- We asses the performance of the proposed scheme for generative modeling tasks with respect to state-of-the-art schemes and for application to uncertainty quantification in inverse problems.

This paper is organized as follows. Section 2 introduces the background and related work. We present for the proposed divergence-free continuous normalizing flows in Section 3. Section 4 introduces the resulting neural architecture and associated learning scheme. We report numerical experiments in Section 5 and further discuss our main contributions in Section 6. We include Proofs in Appendix.

Background and related work

Inverse problem and uncertainty quantification:

As stated in the introduction, we address uncertainty quantification issues in inverse problems. Let us assume that we are provided with a reconstruction scheme $\mu(Y)$ for state X given partial observation Y. As mentioned above, we may consider both model-driven and learning-based schemes for the design of $\mu(\cdot)$. Let $\varepsilon(X)$ be the reconstruction error:

$$\varepsilon(X) := Y - \mu(X) \tag{4}$$

We aim at characterizing the density function $p(.|y)$ associated to the reconstruction error ε $p(.|y)$ and it's associated differential Shannon entropy. When considering Gaussian approximations for posterior $p(x|y)$, one can derive analytically this quantity. It is however widely acknowledged that this approximation does not apply to most real-world processes which involve non-Gaussian statistics (Evensen 2009). State-of-the-art schemes also exploit Monte Carlo methods when one can sample posterior $p(x|y)$ and compute log-likelihood $\log(p(\varepsilon(Y)))$ The latter opens the floor to the exploitation of learning-based generative models such as normalizing flows (Rezende and Mohamed 2015).

Learning-based inference and normalizing flows:

From a probabilistic viewpoint, the maximum likelihood estimation (MLE) is the classic formulation for inference of probabilistic models. Given a parametric family of distributions according to parameter set Θ, we aim at identifying parameters θ^* which maximize the likelihood of the reconstruction error $\varepsilon(Y)$ conditionnally to observation Y. This leads to the following minimization issue:

$$\theta^* = \arg\max_{\theta\in\Theta} \sum_{i=1}^{N} log(p(x_i - \mu(y_i)|y_i, \theta)) \tag{5}$$

Deep learning has recently widened the class of parameteric probabilistic models one can consider in MLE framework through the introduction of normalizing flows. Normalizing flows (Rezende and Mohamed 2015; Kobyzev, Prince, and Brubaker 2020) make use of the formulas the change of variables to extend the MLE framework beyond classic parametric probabilistic families. Given a latent random variable u of density p, the density $\hat{p}$ of the image $v = f_\Phi(u)$ is given by:

$$\hat{p}(v) = \frac{1}{|Jf_\Phi(u)|} p(u) \tag{6}$$

where $Jf_\Phi(u)$ denotes the Jacobian matrix of f_Φ evaluated at point u. Given a parameterization of the likelihood of the latent variable, usually as a Gaussian distribution, normalizing flows then come to design mapping operator f_Φ so that one can compute the determinant of its Jacobian.

Among normalizing flow approaches, continuous normalizing flows (CNF) (Chen et al. 2018) states mapping operator F_Φ as a flow operator governed by an ordinary differential equation:

$$\frac{\partial u(t)}{\partial t} = f_\Phi(u(t)) \tag{7}$$

Under this parameterization, the resulting mapping for likelihood values involve the divergence of the flow:

$$\frac{\partial log(p(u(t))}{\partial t} = -Tr\left(\frac{\partial f_\Phi(u(t))}{\partial u(t)}\right) = div\left(f_\Phi\right) \tag{8}$$

This equation relates to a Lagrangian point of view of Liouville equations which describe the infinitesimal evolution of the log-density by a flow governed by the ODE $\frac{\partial u}{\partial t} = f_\Phi(u(t))$. This framework allows to fit complex distributions without computing the determinant of (6) which is $O(N^3)$. The price to pay is the numerical integration of the ODE.

Equation (8) also stresses that divergence-free flows satisfying $div(f_\Phi) = 0$ would even simplify the computation of likelihoods to the evaluation of the likelihoods in the latent space. We explore this avenue in this work and show it also leads to an analytical derivation of the differential Shannon entropy for image variable v. While the parameterization of divergent-free flows is relatively straightforward for 2D and 3D processes using Helmoltz decomposition of velocity fields, we also address higher-dimensional processes. We may also point out that divergent-free flows also relate to

some extent to Halmitonian flows. This class of flows is governed by some underlying energy functions. Recent exploration of trainable Halmitonian flows (Greydanus, Dzamba, and Yosinski 2019; Sanchez-Gonzalez et al. 2019) for the data-driven discovery of dynamical systems emphasize the relevance of constrained neural ODE schemes to regularize learning problems.

Divergence-free continuous normalizing flow

Here, we focus on a particular class of continuous normalizing flow by enforcing a divergence-free constraint on the CNF. This constraint relates to an incompressibility or volume-preserving constraint on the vector fields defined by the CNF. We show that the proposed divergence-free CNF allows us to derive an analytical computation of information-theoretic quantities such as the differential Shannon entropy. Formally, let us consider the following CNF:

$$\begin{cases} u(0) & \sim \mathcal{N}(\mu,\, \sigma^2) \\ \dfrac{\partial u(t)}{\partial t} & = f_\Phi(u(t)) \;\; \textbf{s.t.}\; div\left[f_\Phi\left(u(t)\right)\right] = 0 \end{cases} \tag{9}$$

Early litterature on normalizing flow already investigated the use of volume preserving flow (Dinh, Krueger, and Bengio 2014) with additive transformation whose triangular matrix is triangular making the computation of its determinant trivial. the proposed approach allows the computation of more complex flows.

Parameterization of divergence-free flows The construction of the particular class of divergence-free normalizing flow presented here rely on the ability to represent divergent-free functions. Let us introduce the following parameterization:

$$f_\Phi = A.\nabla\Psi \tag{10}$$

where A is an antisymmetric linear operator satisfying $A^T = -A$ and Ψ a learnable scalar function. Overall, parameters Φ combine the parameterization of matrix A and of scalar function Ψ. As detailed below, functions of the form are divergent free. For a more intuitive physical interpretation of the presented approach, we adopt a graph representation of the function described above.

Graph representation of divergence-free flows: For any point of $\mathbf{R}^n$, any governing ODE $x'(t) = f(x(t))$ is described by a graph (V, E) and a potential function Ψ. For any edge $(x_i, x_j) \in E$, we associate the stream function defined as

$$S_{ij}(x) = [0, ..., -\frac{\partial\Psi(x)}{\partial x_j}, 0, .., \frac{\partial\Psi(x)}{\partial x_i}, 0, ...0] \tag{11}$$

whose $i-th$ component equals $-\frac{\partial\Psi(x)}{\partial x_j}$ and $j-th$ component equals $\frac{\partial\Psi(x)}{\partial x_i}$. For each $(x_i, x_j) \in E$ we then associate a weight w_{ij}, the dynamical model f is stated as :

$$f(x) := \sum_{(i,j)\in E} w_{ij} S_{ij}(x) \tag{12}$$

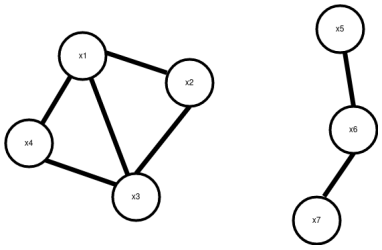

Figure 1: An example of graph (V,E) with two components : nodes represents the data features, edges their connection through the existence of a function S_{ij} described below. In terms of physical interpretation the existence of an edge ij authorize the learned model to advect the measure in the plan of $\mathbb{R}^2$ corresponding to features x_i, x_j

The graph (V, E) describes the coupling between data features which in turn relates to ODE $x'(t) = f(x(t))$. The following lemma state that f can be split onto independent dynamic corresponding to components of (V, E) which are divergent-free:

Lemma 1. *Let assume that the graph (V, E) is composed of N components (V_n, E_n). Then the equation $x'(t) = f(x(t))$ can be split into N coupled non-autonomous equations $x'_n(t) = F_n(x(t))$ with F_n verifying :*

$$Tr(\nabla_{X_k} F_k(x)) = 0 \tag{13}$$

As a corollary, we retrieve the expected divergent-free feature for the considered parameterization.

Corollary 2. *The function f described in (Eq.11) is divergent-free i.e :*

$$div(f) = 0 \tag{14}$$

From this property, we can derive the volume-preserving property of the considered CNF:

Lemma 3. *Let f be a $C^1(\mathbb{R}^n, \mathbb{R}^n)$ function. Then, for any $t \in \mathbb{R}^+$, the semi-group $\Gamma(t)$ of solutions of the ODE : $x'(t) = f_\Phi(x(t))$ is volume preserving i.e :*

$$|J\Gamma(t)(x)| = 1 \tag{15}$$

for any $x \in \mathbb{R}^n$ where $J\Gamma$ denotes the jacobian of flow Γ.

Robustness to mode collapse: Training continuous normalizing flows under the maximum likelihood framework may be subject to generate probability density function with singularities. Let think about a dynamical system f for which each training point x_i is attractive. This would result in the negative likelihood function going to $+\infty$. From a dynamical system perspective, a point x_i is attractive if $f(x_i) = 0$ and if the eigenvalues of the linearized system have negative real-part eigenvalues. As stated by the following theorem, the considered CNF has no attractive points, which in turn should prevent from mode collapse issues.

Theorem 4 (Robustness to mode collapse)**.** *Let $x \in \mathbb{R}^n$ be a zero of the divergent-free function described above i.e $f\Phi(x) = 0$. Then the eigenvalues of $Df(x)$ have zero real parts.*

Derivation of information-theoretic quantities

We further exploit the key features of the proposed CNF to derive information-theoretic quantities. More specifically, the flow defined by (Eq. 9) preserves the differential Shannon entropy as a direct by-product of the divergence-free constraint.

Lemma 5. *Let be q a $C^1(\mathbb{R}^n, \mathbb{R})$ scalar function, and Q the scalar quantity defined as :*

$$Q(p) = \int_{\mathbb{R}^n} q(p(x))dx \tag{16}$$

If $\Phi \in Diff(\mathbb{R}^n)$ denotes a diffeomorphism of $\mathbb{R}^n$ which satisfies $|J\Phi(x)| = 1$. Then, the push-forward density $\tilde{p} := p\#\Phi$ verifies

$$Q(p) = Q(\tilde{p}) \tag{17}$$

In particular, p and $\tilde{p}$ have same differential Shannon entropy.

As the divergence-free constraint implies the volume-preserving property, the later applies to the proposed CNF to derive information-theoretic quantities. More specifically, the flow defined by (Eq.9). A direct consequence of this lemma is that the differential Shannon entropy of any push-forward density image according to (Eq.9) is solely determined by the entropy of initial distribution $\mathcal{N}(\mu, \sigma^2)$ given by $\mathcal{H} = \frac{1}{2}ln\{(2\pi e)^d det(\sigma^2)\}$. As such, it avoids computationally-expensive computation required by classical Monte-Carlo methods which integrates the high-dimensional ODE for each sample.

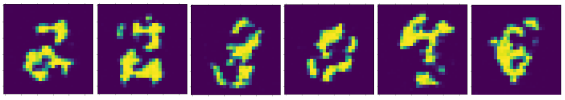

Figure 2: Samples generated from a diagonal gaussian after 2 days of training : The divergence-free flow was limited to a 8-neighbours graph structure, it fail to generate realistic samples, however we discuss in appendix of way to improve the expressiveness of the presented approach.

Learning framework

Given an arbitrary algorithms $\mu(.)$ which delivers a reconstruction of state X from partial observations Y, we address the characterization of the reconstruction error using a neural implementation of the proposed divergence-free CNF. The proposed implementation relies on CNN architectures satisfying the divergent-free constraints presented above. As stated in (Eq.11), our model relies on learning a CNN parameterization for potential Ψ and a linear antisymmetric transformation. We first describe the latter. We then introduce the resulting end-to-end neural architectures and the associated training setting.

The antisymmetric transformation layer: In order to apply the proposed framework to high-dimensional n-dimensional space, we propose to implement a translation-invariant antisymmetric transformation in a CNN-fashion.

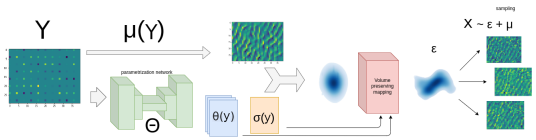

Figure 4: Given an arbitrary denoising algorithms μ, the proposed neural network architecture aim to learn the density function p associated to the reconstruction error ε : first a CNN neural network Θ (green) extract both features θ and log-variance $ln(\sigma)$ of an initial gaussian distribution, then a divergence-free flow parametrized by θ advect the gaussian distribution using an ODE solver (red) to match the reconstruction error distribution ε in the sense of maximum-likelihood estimation.

This can simply be done using convolutional transform and applying a antisymmetric operator to the channel dimension. as described in Figure 3.

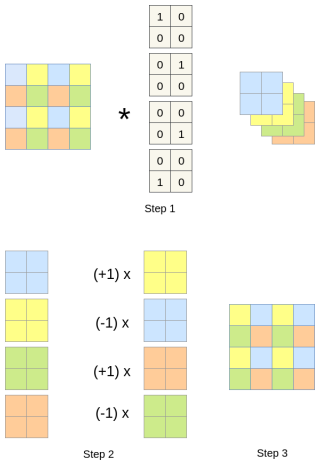

Figure 3: For 1-channel image, the antisymmetric transform layer consist in 3-step. Step 1 : a fixed convolution layer is applied to an input, which are stacked onto 4 channels. Step 2 : a antisymmetric operator is applied over the channel. Step 3 : the inverse of operation performed in step 1.

Neural architecture : overall, the considered neural architecture for the modeling and characterization of the posterior $p(X|y)$ is sketched in Fig.4. The considered neural architecture involves two main components:

- A first neural network Θ which takes as inputs the observation Y and the reconstructed state $\mu(X)$ and outputs the initial condition of the flow. It combines a prediction of the the covariance of the initial Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$ (we consider a diagonal coavariance parameterization); z_0: an augmented component used by the neural integrator to better advect the initial gaussian density. We may recall that the estimation of the mean $\mu(x)$ is given by an arbitrary denoising algorithm μ.
- The second neural network consists in a Runge-Kutta integration scheme ofof the considered divergent-free

neural flow f_Φ applied to initial condition $u(0) = (\varepsilon(X), z_0)$. For the numerical integration, we make use of the open source library *torchdiffeq* provided by the authors of (Chen et al. 2018).

Training setting :

Given the considered neural architecture, we train the proposed divergence-free CNF using a MLE criterion (Eq.**??**) as considered in (Chen et al. 2018; Grathwohl et al. 2018):

$$\mathcal{L}(\theta) = \sum ln(|\sigma|) + (z_i - \mu(\hat{x}))^t \sigma^{-1}(z_i - \mu(\hat{x})) \quad (18)$$

with $z_i := \Phi(-t)(y_i, \Theta(\hat{x}))$ the output of the flow integrated backward in time. We also perform training of a gaussian model with full covariance for benchmarking purpose.

All experiments were run using pytorch. The training procedure is performed by gradient descent using Adam algorithm over 150 epoch. As mentioned above, the neural solver for ODE is the pytorch implementation of (Chen et al. 2018).

Numerical experiments

This section reports numerical experiments for the proposed approach. We first assess divergence-free CNF for generative modelling tasks before considering applications to inverse problems and uncertainty quantification.

Benchmarking for generative modeling tasks : We perform density estimation tasks using the benchmarking framework proposed in (Papamakarios, Pavlakou, and Murray 2017). The reported quantitative comparison involves state-of-the-art normalizing flow approaches, namely the masked autoregressive flows (MAF) (Papamakarios, Pavlakou, and Murray 2017), the masked autoencoder for distribution estimation (MADE) from (Germain et al. 2015) and the continuous normalizing flow.

method	POWER	GAS	HEPMASS	MNIST
Gaussian	-7.74	-3.58	-27.93	-1366
MADE	-3.08	3.56	-20.98	-1380
FFJORD	**-0.46**	**8.59**	**-14.92**	NA*
MAF	0.14	9.07	-17.7	-1300
VPCNF (ours)	-1.96	6.32	-20.13	-1176

Table 1: Caption : performances in terms of likelihood over the test set (higher is better).* : authors provides results in terms of bits per dim

All schemes are evaluated in terms of log-likelihood of the test set. We report state-of-the-art performance, better than MADE (Germain et al. 2015), with a clear improvement compared with the Gaussian baseline. Given that the proposed scheme can be regarded as a constrained version of (Grathwohl et al. 2018), we do not expect to outperform (Grathwohl et al. 2018) in general. Sampled examples illustrated in Fig.2 for MNIST dataset may in this respect point out that we may increase the complexity of the our implementation to sample more realistic examples. We may however stress that the considered parameterization was primarily considered here for an application to unvertainty quantification as illustrated below.

Reconstruction of Lorenz-63 dynamics

We illustrate the relevance of the proposed framework for uncertainty quantification in inverse problems with dynamical systems. As toy model, we consider Lorenz-63 system which involve chaotic dynamics:

$$\begin{array}{rcl} x_1'(t) & = & \sigma(x_2 - x_1) \\ x_2'(t) & = & x_1(\rho - x_3) - x_2 \\ x_3'(t) & = & x_1 x_2 - \beta x_3 \end{array} \quad (19)$$

We asses the ability to estimate the conditional probability $p(x_T|y_{1:T})$ of the state x_T given all previous noisy observations $Y_{1:T-1}$. We state the observational model as $H(x_1, x_2, x_3) = x_1 + x_2$. Given a time series of noisy observations, a LSTM (μ) is trained to reconstruct the true state while an other LSTM (Θ) is trained to estimate the parameters of the flow : In (de Bezenac Emmanuel 2020) authors propose the same framework for filtering, the estimation of the conditional distribution was performed using realNVP (Dinh, Sohl-Dickstein, and Bengio 2016). We report results in Table 2 where we compare the learned model with a Gaussian baseline. They clearly point out that the proposed scheme outperforms a Gaussian approximation for the posterior, which is the classically considered including in learning-based variational Bayesian scheme (Kingma and Welling 2013). Figure 7 further reveals the non-Gaussian features of the posterior when the state is close to the bifurcation zone from one lobe to another one of Lorenz-63's attractor. This Figure plots samples of the inferred posterior $p(X_T|Y_{1:T})$ for different values of T for a true state close to the bifurcation zone. As expected, the more observations, i.e. T being larger, the lower the spread of the samples, which indicates a lower uncertainty. Figure 5 further illustrates this global pattern through the comparison of the inferred Shannon entropy averaged over the test set compared with the mean square error of the reconstruction. We may recall that we cannot derive analytically the true posterior, nor true Shannon entropy for Lorenz-63 dynamics. We further illustrate the relevance of the proposed derivation of the Shannon entropy of the posterior in Figure 6. The inferred entropy alongside the attractor of the Lorenz 63 nicely emphasizes that the reconstruction uncertainty is larger when getting closer to the bifurcation zone.

prior	T = 1	T = 3	T = 5	T = 7	T = 12
gaussian	-0.47	2.27	3.34	4.22	4.34
VPCNF	**0.43**	**3.35**	**4.39**	**4.67**	**4.98**

Table 2: Reconstruction performance in terms of conditional log-likelihood over the test set for Lorenz-63 case-study: we compare the proposed approach to a Gaussian baseline. We refer the reader to the main text for the experimental setting.

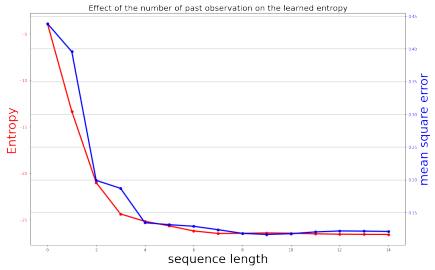

Figure 5: Evolution of the mean differential Shannon entropy associated to the inferered posterior $p(X_T|Y_{1::T})$ with respect to time horizon T for Lorenz-63 case-study (red). For comparison purposes, we consider the evolution of the mean square error (MSE) of the reconstruction of the true state (blue).

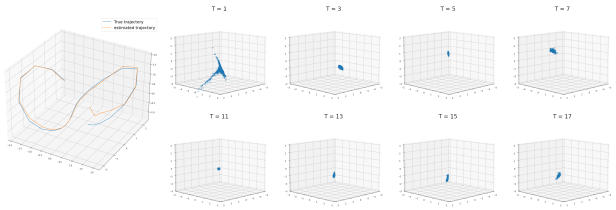

Figure 7: Left : true and inferred Lorenz-63 states and states, right : sampling of 1000 points of the inferred posterior $p(x_T|y_{1:T})$ for different value of T. It clearly illustrates the ability of the proposed framework to address non-Gaussian features.

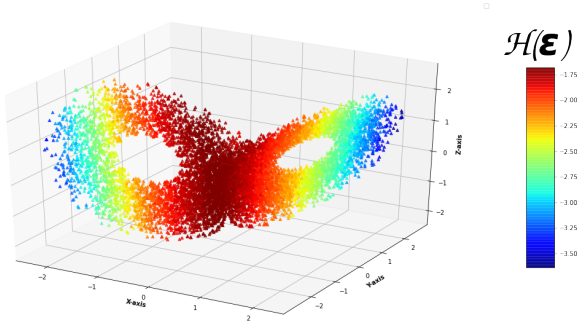

Figure 6: Differential entropy of the estimated distribution $p(\varepsilon|y_1)$ over the test dataset :As expected, the entropy of the estimated density is higher over the middle region where it is hard to predict in which part of the attractor the true state lies due to the chaotic nature of the governing equations.

MNIST case-study : We consider an inpainting application for MNIST dataset and aims at assessing how the observation patterns impact the reconstruction performance. We set two observation configurations with the same number of observed pixels set to $d = 81$: the first configuration involves regularly-sampled pixels in both horizontal and vertical directions, the second configuration samples pixels regularly-spaced along the horizontal axis and randomly along the vertical axis. We also add a gaussian white noise to the data to ensure that the reconstruction error does not lie in an area with zero Lebesgue measure. We trained two neural network μ_1, μ_2 on the image space to perform the reconstruction from the gappy observations. For each reconstruction network, we train a divergence-free CNF to model the reconstruction posterior and evaluate the associated Shannon entropy. We reduce train and test sets to digits from 7 to 9 in order to reduce training time and power consumption of the experience. We perform training over 600 epoch using Adam algorithm and stop each models after 36 hours of training, no improvement over the training set were observed. We report in Fig. 8 examples of the random sapling patterns, of mean reconstruction and samples from the inferred posterior for the randomly-sampled observations. We report in Fig.9 the distributions of reconstruction error of the inferred Shannon entropy for the two observation configurations. The random sampling configuration leads on average to smaller reconstruction errors, our model infers more similar distribution of reconstruction uncertainty, with even a greater uncertainty for the iregular sampling configuration. This is in line with compressive sampling results. While random sampling strategies improve almost surely the reconstruction performance compared with a regularly-spaced sampling, it may also lead to reconstruction outliers, which in turn results in heavier-tail for the posterior distribution. We interpret the results reported in Fig.9 as an illustration of the proposed scheme to capture differences in the non-Gaussian tails of the posterior through the inferred Shannon entropy.

Figure 8: left : masked input y, center : estimated reconstruction $\mu(y)$, right : a data sample s generated using the learned distribution $\hat{\varepsilon}$ by the additive relation $s = \mu(y) + \hat{\varepsilon}$

Conclusion

We introduce a class of divergence-free continuous normalizing flows with a specific interest in uncertainty quantification for inverse problems. Being associated with volume-preserving features, these flows lead to an analytical derivation of information-theoretic quantities for the inferred posterior, especially the conditional Shannon entropy. As such, it extends mathematical results on the estimation of the differential Shannon entropy of complex distributions such as Gaussian mixture to a wider class of probability density function. Numerical experiments support state-of-the-performance for generative modelling tasks compared to classic normalizing flows. The proposed framework opens new avenues for using Shannon entropy criterions for uncertainty quantification and inverse problems, especially regarding the comparison of observing systems or sensor placement problems. Beyond such applications for real-world systems, future work may further investigate

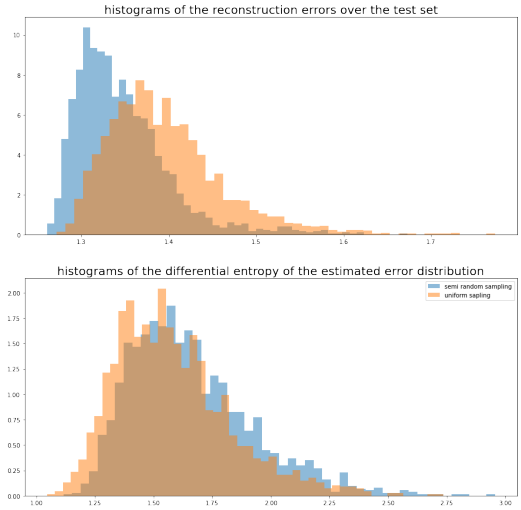

Figure 9: top : Histograms of the reconstruction error over the test set for the two different scenarios (blue : random samplings, orange : uniform samplings), bottom : Histogram of the differential entropy of the estimated reconstruction error distribution.

how volume-preserving transformations and the underlying graph structure of the dynamical system affects the learning process.

Appendix

proof of lemma 1 Suppose the graph (V, E) is composed of N components (V_n, E_n) : then the equation $x'(t) = f(x(t))$ can be split onto N coupled non-autonomous equations $x'_n(t) = F_n(x(t))$ with F_n verifying :

$$Tr(\nabla_{X_k} F_k(x)) = 0 \tag{20}$$

Proof. Without loss a generality, we suppose $x = [X_1, X_2, ..., X_N]$ and $f = [F_1, ..., F_N]$.

$$\begin{aligned} Tr(\nabla_{X_n} F_n(x)) &= \sum_{k \in V_n} \frac{\partial f_k}{\partial x_k} \\ &\sum_{k \in V_n} \Big(\sum_{(ij) \in E_n} w_{ij} \frac{\partial S_{ij(x)_k}}{\partial x_k} \Big) \\ &\sum_{(ij) \in E_n} w_{ij} \sum_{k \in V_n} \frac{\partial S_{ij(x)_k}}{\partial x_k} \\ \sum_{(ij) \in E_n} w_{ij} \Big(\sum_{k \neq i,j} 0 &+ \frac{\partial S_{ij(x)_i}}{\partial x_i} + \frac{\partial S_{ij(x)_j}}{\partial x_j} \Big) \\ = \sum_{k \in V_n} \sum_{k \neq i,j} 0 &+ \frac{-\partial^2 \Psi(x)}{\partial^2 x_i x_j} + \frac{\partial^2 \Psi(x)}{\partial^2 x_j x_i} \\ &= 0 \end{aligned} \tag{21}$$

The last line follow from the Cauchy Lemma □

Proof of lemma 4:

Proof. if we denote as $z = \Phi(x)$, the change of variable formula states that $\tilde{p}(z)$, the image density is given by $\tilde{p}(z) = \frac{p(x)}{|J\Phi(x)|}$

$$Q(\tilde{p}) = \int q(\tilde{p}(z))dz = \int q(\frac{p(x)}{|J\Phi(x)|})|J\Phi(x)|dz \tag{22}$$

It follows from $dz = |J\Phi(x)|dx$ and $|J\Phi(x)| = 1$ that :

$$Q(\tilde{p}) = \int q(\tilde{p}(z)dz = \int q(p(x))dx = Q(p) \tag{23}$$

Thus, the differential Shanon entropy $\mathcal{H}(\tilde{p})$ of an image density associed to a measure-preserving flow is invariant. It is entierly determined by the initial density p and ca be computed analytically if $\mathcal{H}(p)$ does. □

Proof of theorem 4:

Proof. if $f = A.\nabla\Psi$, then $Df(x) = A.\nabla^2\Psi$ which is antisymetric as the matricial product between the antisymmetric matrix A and the symmetric hessian matrix $\nabla^2\Psi(x)$. Thus its eigenvalues have zero real part. □

limitations of the proposed approach

Given a divergent free flow $\mathcal{F}$ and a famillly $\mathcal{P}_0$ of initial probability density, It may arise that the presented framework here still lack of expressiveness in order to fit an arbitrary density probability f. For example, the set of 1-dimensional divergent free flow contains only the translations. Here we discuss about two ways to overcome this issue :

Graph structure and data representation : All experience performed in this paper (except for the Lorenz-96) make use of the same 8-neighbours graph-structure as described in figure 10. Because it remains fixed during the training, this may limit the performances of the proposed framework for density estimation tasks. In future works we may investigate the optimization of this graph and consider different representation of the input. In (Voleti et al. 2021) authors provides an interesting multi-scale representation using volume preserving transformations.

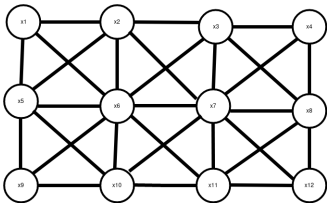

Figure 10: 8-Neighbours graph-structure with $n \times m$ vertices used in numerical experiments presented above. It have a number of edges $\#E = 4nm - 3(n + m) + 2$.

Augmented state Given a set of examples $x_i \in \mathbb{R}^n$, we can construct an a set of augmented variable $\tilde{x_i} := [x_i, \varepsilon_i]$ with $\varepsilon_i \sim \mathcal{N}(0, \sigma I)$ and apply the above mentioned framework This may help to match an arbitrary distribution f with the marginal density of the learned probability density. The price to pay is that the Shanon entropy of the marginal density of f is no more given by the initial density, we can only compute a lower bound given the inequality :

$$H(X, Y) \leq H(X) + H(Y) \tag{24}$$

Extend the set $\mathcal{P}_0$ In this work we only considered diagonal gaussian distributions as set $\mathcal{P}_0$ of initial density function, the action of volume preserving on this set results in a non-universal approximator. However, it extends any mathematical results on estimation of the differential Shanon entropy of any initial set $\mathcal{P}$.

References

Ao, Z.; and Li, J. 2022. Entropy estimation via normalizing flow. *sign*, 30: 1.

Bishop, C. 2006. Pattern recognition and machine learning.

Chen, R. T.; Rubanova, Y.; Bettencourt, J.; and Duvenaud, D. K. 2018. Neural ordinary differential equations. *Advances in neural information processing systems*, 31.

Courtier, P.; and Talagrand. 1987. Variational Assimilation of Meteorological Observations With the Adjoint Vorticity Equation. Ii: Numerical Results. *Quarterly Journal of the Royal Meteorological Society*.

de Bezenac Emmanuel. 2020. Normalizing Kalman Filters for Multivariate Time Series Analysis. *NIPS*.

Dinh, L.; Krueger, D.; and Bengio, Y. 2014. Nice: Nonlinear independent components estimation. *arXiv preprint arXiv:1410.8516*.

Dinh, L.; Sohl-Dickstein, J.; and Bengio, S. 2016. Density estimation using real nvp. *ICLR 2017*.

Dong, C.; Loy, C. C.; He, K.; and Tang, X. 2014. Learning a deep convolutional network for image super-resolution. In *European conference on computer vision*, 184–199. Springer.

Evensen, G. 2009. *Data Assimilation*. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN 978-3-642-03710-8 978-3-642-03711-5.

Fablet, R.; Chapron, B.; Drumetz, L.; Memin, E.; Pannekoucke, O.; and Rousseau, F. 2021. Learning Variational Data Assimilation Models and Solvers. *JAMES*. ArXiv: 2007.12941.

Germain, M.; Gregor, K.; Murray, I.; and Larochelle, H. 2015. Made: Masked autoencoder for distribution estimation. In *International conference on machine learning*, 881–889. PMLR.

Grathwohl, W.; Chen, R. T.; Bettencourt, J.; Sutskever, I.; and Duvenaud, D. 2018. Ffjord: Free-form continuous dynamics for scalable reversible generative models. *ICLR 2018*.

Greydanus, S.; Dzamba, M.; and Yosinski, J. 2019. Hamiltonian neural networks. *Advances in neural information processing systems*, 32.

Huber, M. F.; Bailey, T.; Durrant-Whyte, H.; and Hanebeck, U. D. 2008. On entropy approximation for Gaussian mixture random vectors. In *2008 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 181–188. IEEE.

Hyvärinen, A. 1997. New approximations of differential entropy for independent component analysis and projection pursuit. *Advances in neural information processing systems*, 10.

James, F. 1980. Monte Carlo theory and practice. *Reports on progress in Physics*, 43(9): 1145.

Kingma, D. P.; and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Kobyzev, I.; Prince, S. J.; and Brubaker, M. A. 2020. Normalizing flows: An introduction and review of current methods. *IEEE transactions on pattern analysis and machine intelligence*, 43(11): 3964–3979.

Minaee, S.; Boykov, Y. Y.; Porikli, F.; Plaza, A. J.; Kehtarnavaz, N.; and Terzopoulos, D. 2021. Image segmentation using deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*.

Nonnenmacher, M.; and Greenberg, D. 2021. Deep Emulators for Differentiation, Forecasting, and Parametrization in Earth Science Simulators. *JAMES*, 13(7): e2021MS002554. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2021MS002554.

Papamakarios, G.; Pavlakou, T.; and Murray, I. 2017. Masked autoregressive flow for density estimation. *Advances in neural information processing systems*, 30.

Rezende, D.; and Mohamed, S. 2015. Variational inference with normalizing flows. In *International conference on machine learning*, 1530–1538. PMLR.

Sanchez-Gonzalez, A.; Bapst, V.; Cranmer, K.; and Battaglia, P. 2019. Hamiltonian graph networks with ode integrators. *arXiv preprint arXiv:1909.12790*.

Shannon, C. E.; and Weaver, W. 1949. *The Mathematical Theory of Communication*. Urbana, IL: University of Illinois Press. ISBN 978-0-252-72548-7.

Voleti, V.; Finlay, C.; Oberman, A. M.; and Pal, C. J. 2021. Multi-Resolution Continuous Normalizing Flows. *CoRR*, abs/2106.08462.

Xie, J.; Xu, L.; and Chen, E. 2012. Image Denoising and Inpainting with Deep Neural Networks. In *NIPS*, 341–349.

Yin, T.; Yuen, K.-V.; Lam, H.-F.; and Zhu, H.-p. 2017. Entropy-based optimal sensor placement for model identification of periodic structures endowed with bolted joints. *Computer-Aided Civil and Infrastructure Engineering*, 32(12): 1007–1024.

Yu, J.; Lin, Z.; Yang, J.; Shen, X.; Lu, X.; and Huang, T. S. 2018. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5505–5514.