PERSPECTIVE • OPEN ACCESS

Data reconstruction for complex flows using AI: Recent progress, obstacles, and perspectives

To cite this article: Michele Buzzicotti 2023 *EPL* **142** 23001

A Letters Journal Exploring the Frontiers of Physics

April 2023

EPL, **142** (2023) 23001
doi: 10.1209/0295-5075/acc88c

www.epljournal.org

Perspective

Data reconstruction for complex flows using AI: Recent progress, obstacles, and perspectives

Michele Buzzicotti[(a)]

Department of Physics and INFN, University of Rome "Tor Vergata" - Via della Ricerca Scientifica 1, 00133, Rome, Italy

received 16 March 2023; accepted in final form 29 March 2023
published online 11 April 2023

Abstract – In recent years the fluid mechanics community has been intensely focused on pursuing solutions to its long-standing open problems by exploiting the new machine learning (ML) approaches. The exchange between ML and fluid mechanics is bringing important paybacks in both directions. The first is benefiting from new physics-inspired ML methods and a scientific playground to perform quantitative benchmarks, whilst the latter has been open to a large set of new tools inherently well suited to deal with big data, flexible in scope, and capable of revealing unknown correlations. A special case is the problem of modeling missing information of partially observable systems. The aim of this paper is to review some of the ML algorithms that are playing an important role in the current developments in this field, to uncover potential avenues, and to discuss the open challenges for applications to fluid mechanics.

Introduction. – There is no doubt that our ability to produce, collect and analyze data is rapidly increasing boosted by a positive feedback loop between technological progress and new algorithms. Computer scientists, engineers, as well as physicists and mathematicians, are pushing toward the new machine learning (ML) era, which has already resulted in reforming standard data analysis paradigms. Breakthroughs have been achieved in numerous areas of computer science, from computer vision (CV) [1,2], up to natural language processing [3–5] as well as in some scientific contexts as the protein folding problem [6].

In *complex flows* as well, there have been numerous positive outcomes in nearly all testing scenarios, varying from control problems as single and multi-agents navigation in complex environments [7–15], to turbulent control and drag reduction [16–21], up to data assimilation problems [22–33] to cite few of them. However, applications in fluids are still in their infancy, and the majority of cases are either conducted on highly idealized setups or only showing preliminary results on more realistic conditions. The objective of this paper is to examine some of the ML tools that have been applied with promising results to reconstruct data from incomplete observations of complex systems, including idealized turbulent [34–41], engineering [42–45], and geophysical flows [32,33], and to discuss the possible future directions for quantitative advancements in fluid mechanics.

Data reconstruction is the art of filling in missing information by interpolating, denoising, or super-resolving a single realization, or a time series, of data fitting a specific statistical distribution [46,47]. ML applications for data reconstruction are emerging in many areas, from computer vision [48–51], to medical imaging [52–54] up to seismic data reconstruction [55,56] and astrophysics [57,58]. Also in geophysical fluid dynamics works using ML to reconstruct missing data are rapidly growing [59–64]. For our focus on reconstructing complex flows it is possible to distinguish four possible different questions, see fig. 1: (i) full-state restoration, with the aim to fill missing gaps in the real space state of a complex flow, (ii) inferring missing fields, which can be derived as the inverse problem solution where a physical observable that cannot be accessed/measured directly can be inferred by measuring other quantities to which it is coupled,

[(a)] E-mail: michele.buzzicotti@roma2.infn.it (corresponding author)

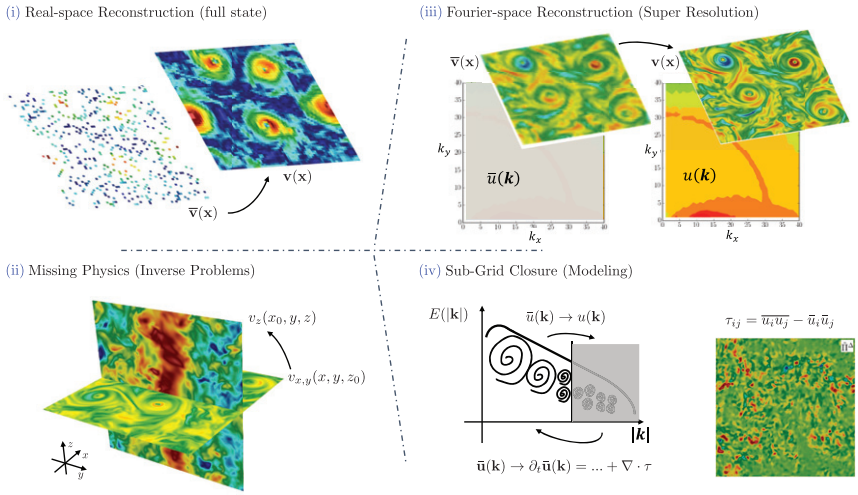

Fig. 1: Graphical illustration of different types of reconstructions. Panel (i): the full state reconstruction from partial observations, gap fill. Panel (ii): inverse problems, reconstruction of missing physical quantities coupled to the observed ones. Panel (iii): super-resolution, this is equivalent to a gap reconstruction on the high Fourier space frequencies. Panel (iv): dynamical reconstruction, or modeling of missing physics on the observed scales.

(iii) super-resolution, which can be seen as the equivalent of point (i) but when the gap to fill is on the high wave numbers of the Fourier domain, (iv) dynamical modeling, which consists of reconstructing dynamically the effects of missing scales on the evolution of the resolved ones [65–68]. The issue of designing a ML-inspired subgrid closure for modeling computational fluid dynamics is a subject per se and has been recently reviewed in [69,70]. Here, our focus lies on the first three categories of problems under the assumption that the amount of missing information to fill in is very large, which renders the problem ill-posed. This means that multiple solutions can fit within the same reconstruction [38,43]. Under this assumption, already defining what the optimal solution is, it is a question that can have different answers depending on the specific target. For instance, as discussed in [71,72], the optimal reconstruction providing the minimum mean squared error (MSE) is different from optimal solutions in terms of other statistical quantities. *In this review, we target reconstructions that maximize the correlations with the observed data while respecting the statistical features of the ground truth solution.* The large-gap assumption is required when dealing with the reconstruction of complex flows. For instance, let us consider the full-state reconstruction problem of ocean surface currents. Even though satellites have allowed us to get, for the first time, a global picture of the ocean [73,74], from mesoscale eddies up to western boundary currents, over time scales relevant to climatological studies (decades), observing the full dynamics of the ocean remains a gigantic task [75,76], and requires filling gaps of spatial scales between hundred km up to less than a meter and time-frequency gaps spanning weeks up to turbulent and wave scales (of seconds), which cannot be neglected to explain turbulent stirring, mixing, and all vertical motions

On top of applications, there are *fundamental questions* associated with reconstructing complex flows. What type and quantity of information are required to perform different reconstructions is one open theoretical question, which can be investigated via a reverse engineering approach: different inputs can be passed to the same model to assess the impact on the reconstruction quality. Here, we discuss some of the ML algorithms that are transforming the conventional paradigms of data analysis and that have the potential to facilitate breakthroughs in the field of fluid dynamics. Following a chronological order, we start with an introduction of "Variational Auto-Encoders" (VAEs), "Generative Adversarial Networks" (GANs), and "denoising Diffusion probabilistic Models" (DMs). After we discuss how to combine pure data-driven methods with the physical knowledge at hand and we provide possible future directions in this discipline.

Data-driven methods. – A typical approach to repair missing data in gappy fields, before the rise of ML, was based on *proper orthogonal decomposition* (POD). POD is used to reduce data dimensionality by identifying the dominant patterns in a dataset and representing them using a smaller set of orthogonal basis functions (POD modes, *i.e.*, eigenvectors of the correlation matrix) [77–79]. The same approach has been extensively used also for filling of missing points in geophysical data sets where it takes the name of Empirical Orthogonal Function [80]. Extension of such techniques as the Gappy POD (GPOD) [81] or the Extended POD (EPOD) [82] were derived to repair missing data with minimal MSE solutions, showing results outperforming Kriging interpolation [83]. However, POD-based approaches are limited when dealing with complex multi-scale and non-Gaussian statistics as is the case of turbulent flows. As shown in [84], where they implemented

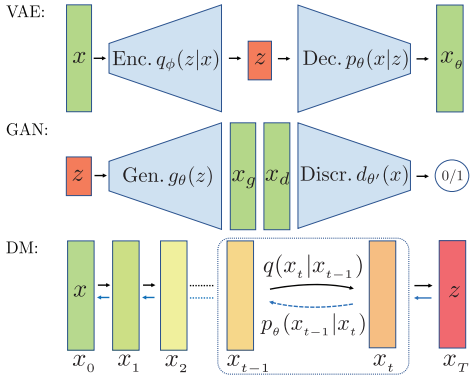

Fig. 2: Schematic representation of the three ML algorithms developed to generate data accordingly with the probability distribution function described by a training dataset. Top: Variational Auto-Encoders, based on a probabilistic encoder $q_\phi(z|x)$ and decoder $p_\theta(x|z)$. Middle: Generative Adversarial Networks, based on a generator $g_\theta(z)$ and a discriminator $d_\theta(x)$. Bottom: Diffusion Models, aiming to model the reverse transition probability of a Markov chain, through a network $p_\theta(x_{t-1}|x_t)$.

EPOD to reconstruct the bulk velocity of wall-bounded turbulence from wall measurements, and in [71] where they used EPOD and GPOD to fill missing data on velocity planes extracted from 3d rotating turbulent flows, POD methods can only reconstruct the large-scale, Gaussian features of the ground truth data.

A significant advancement in this regard was brought by the ML *generative models* [85–87] aiming to generate new data that resembles "statistically" the training dataset. Their success can be attributed to two factors. Firstly, their architecture relies on multi-layer Convolutional Neural Networks (CNN) [88,89], which inherently possess the ability to emphasize long-range correlations in data. Secondly, they are trained with loss functions that not only account for MSE accuracy but also for statistical differences between the generated and ground truth data. Figure 2 gives a schematic illustration of the three main generative models that are commonly utilized in ML.

VAEs have been the first type of neural network trained to give in output new samples, not belonging to the training dataset, which satisfy the same statistical properties. In the first row of fig. 2, a simple diagram depicts the workflow of the VAE model. VAEs, like their predecessors, Auto-Encoders, are based on an encoder-decoder structure. However, unlike Auto-Encoders, the aim of VAEs is not to perform a dimensionality reduction by projecting the input data, x, into a low-dimensional latent space, z. Instead, VAEs define a probabilistic decoder, $p_\theta(x|z)$, which maps any input from the latent space, sampled from a simple distribution, $p(z)$, typically a multivariate Gaussian, into a sample in the output space that satisfies the (generally unknown) statistical distribution characterizing the training dataset. The probabilistic encoder, $q_\phi(z|x)$, plays a crucial role in VAEs by facilitating the sampling of the latent space, z, during training to accelerate decoder convergence. The encoder's primary objective is to model the posterior probability of the decoder, denoted as $p_\theta(z|x)$, which corresponds to the likelihood of obtaining a particular sample in the latent space z when generating a specific input from the dataset x. The presence of the probabilistic encoder assists the discriminator in exploring a smaller and more relevant submanifold of the latent space, resulting in faster and more stable training. VAE models operate on the basic assumption of learning a mapping between a simple and fixed distribution into the data probability distribution. Training the encoder entails minimizing the Kullback-Leibler Divergence (KLD) between the selected latent space distribution, $p(z)$, and the encoder's output distribution, $q_\phi(z|x)$. This operation can generally be computed analytically and requires adding a few extra terms to the decoder loss function. As previously mentioned, the probabilistic decoder of the VAE is trained by maximizing the log-likelihood of the generated data, $\log p_\theta(x)$, where

$$p_\theta(x) = \int p_\theta(x|z)p(z)\mathrm{d}z.$$

Directly computing this loss function is intractable. However, a lower bound can be defined, using a variational inference formulation, and calculated under some approximations. The approximations are based on the assumption that the decoder's errors are Gaussian. By making this assumption, the maximization of the log-likelihood can be rewritten as a minimization of the MSE. While this approximation may be reasonable in some contexts, it is certainly unsuitable for considering turbulent flows, which are well known for their highly non-Gaussian extreme fluctuations. As shown in the context of turbulent flows on a rotating frame [71], the minimization of MSE alone results in generating solutions that match the training data only at the large, more energetic scales, while over-damping the smaller scales. Therefore, rather than as generative methods, VAEs are mostly considered in the context of reduced-order modeling to perform a probabilistic projection on low-dimensional latent space, z, as studied in the context of 3d homogeneous and isotropic turbulent (HIT) flows [34] and more recently on a 2d flow of a simplified urban environment [90].

GANs are proposed to improve VAEs by relaxing the Gaussian errors assumption, and by improving the evaluation of statistical features in generated data in the loss [91–93]. In general, the functional form of the probability distribution that characterizes the training dataset is unknown. To overcome this issue, a second network, the discriminator, $d_{\theta'}(x)$, is used to evaluate the statistical properties of training and generated datasets. The discriminator provides a loss function that the GAN generative part, $g_\theta(z)$, can optimize during training. The discriminator functions as a classifier and is trained to assign a probability of an input being generated or extracted from a true dataset. On the other hand, the generator maps a sample from a latent space into a sample in the data space,

similar to a VAE decoder. Its objective is to generate increasingly realistic samples that can fool the discriminator, from which comes the name "adversarial", where, as in a zero-sum game, a gain for one network gives an equivalent loss to the other [94,95]. For a fixed generator the analytical expression for the optimal discriminator can be derived by maximizing the adversarial loss [91], and results in

$$d^*(x) = \frac{p_{true}(x)}{p_{true}(x) + p_{gen}(x)},$$

where p_{true} and p_{gen} represent the statistical distributions of the true and generated datasets. Similarly, the optimal generator, denoted as $g^*(z)$, can be derived as the network that minimizes the Jensen-Shannon Divergence (JSD), a symmetric formulation of the KLD, between the true and generated distributions [91]. GANs have exhibited unparalleled potential in producing turbulent datasets that display a remarkable level of statistical similarity to their original counterparts. Both the original and generated data exhibit identical deviations from Gaussianity up to the evaluation of high-order statistical observables in several setups, as in super-resolving to a 64× larger 2d turbulent flows behind cylinders [96], and of 3d HIT flows [36,97], as well as in filling large gaps in rotating turbulence [38] and 3d channel flows [43]. Figure 3 showcases the workflows of VAEs and GANs focusing on the applications of these models to fill gaps by exploiting their generative capacities also when constrained to fit some observations. In the three panels (a), (b) and (c), the sample $\hat{x}$ represents the gappy data and serves as a condition to the model, the ground truth data (known only in the training stage) is denoted as x_d, the model reconstruction is called x_g. In the VAEs the condition $\hat{x}$ is passed to both the encoder and the decoder. During training the encoder projects x_d and $\hat{x}$ into the latent space by defining the variance and the mean of a Gaussian distribution from which a sample z is extracted. The loss function is the same as in the case of pure generation. In testing setup, the reconstruction of the sampling on z is done from a standardized multivariate Gaussian while the decoder on top of the z sample analyzes also the condition $\hat{x}$ [98–100]. Panel (b) displays the GAN reconstruction setup, which distinguishes itself from the unconstrained model in that the generator employs an encoder-decoder architecture to map $\hat{x}$ to an intermediate space z prior to generating the filling data instead of performing a random sampling on the latent space. The discriminator operates as usual, but now the overall generation loss is a linear combination of the MSE between the ground truth and the reconstruction data, in addition to the adversarial loss provided by the discriminator prediction [48,71,72]. GAN generates realistic samples also when constrained to match some observations. However, in the reconstruction case having statistically consistent data leads to a larger MSE with respect to the ground truth solutions. Indeed a tiny shift in space between the reconstruction and the true solution brings larger MSE if the fields are both highly fluctuating [71]. The principal

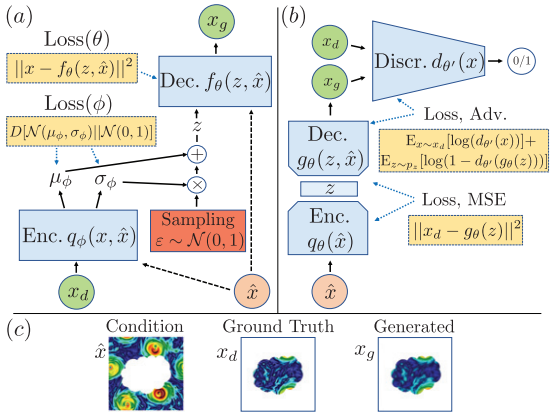

Fig. 3: Workflow of a typical Variational Auto-Encoder (panel (a)) and of a typical Generative Adversarial Network (panel (b)) designed to generate samples x_g conditioned on some observations $\hat{x}$. The gray boxes represent the functions optimized during the training, the yellow boxes report the loss functions and their connection with the different parts of the network. The red box indicates where the stochastic sampling is happening along the network propagation, while the green and brown circles represent, respectively, the input and the conditioning of the networks. (c) Visualization of typical fields analyzed by the networks with the aim of transforming incomplete data into corresponding complete data.

limitation of GANs arises from their adversarial nature, resulting in highly unstable training and slow convergence. It can happen that one of the two players is dominated by the other and converges into a failure solution.

Diffusion Models, fig. 2, bottom panel, are an alternative technique to generate data. DMs transform a simple distribution into a more complex distribution, resembling the training data while avoiding the need to introduce a surrogate loss function, as seen in VAEs, and without relying on adversarial training, as in GANs. The workflow of DMs is illustrated in fig. 4, for both the generation (a) and the reconstruction (b) setups. DMs use a Markov chain to gradually convert one distribution (latent space) into another (dataset), following the idea developed in non-equilibrium statistical physics [101]. To learn the parameterized Markov chain, DMs are trained using variational inference to produce data samples that match the original data after repeating a finite number of steps. The learning involves estimating small perturbations to a diffusion process, a problem which is more tractable than explicitly describing the full distribution within a single jump as potentially done in the other generative models. Furthermore, since a diffusion process, $q(x_t|x_{t-1})$, exists for any smooth target distribution, this method can capture data distributions of arbitrary form [101]. If the forward diffusion process is a Markov chain that gradually introduces Gaussian noise to the data until the signal is destroyed, the model subsequently learns how to reverse the diffusion process and generate desired data samples starting from pure Gaussian noise realizations. Unlike VAEs or GANs, diffusion models involve a latent variable z with

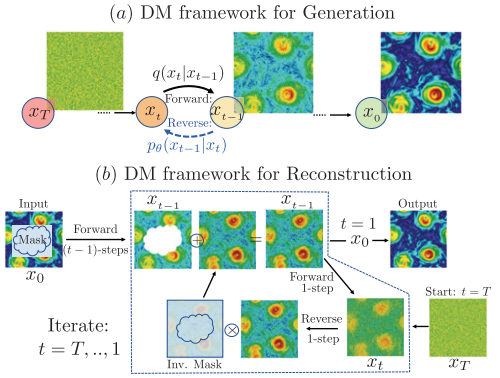

Fig. 4: (a) Typical workflow of a Diffusion Model designed and trained to generate data respecting the statistical distribution of a training dataset. Framework applied to reconstruct the full state (reconstructed output) from partial observations (masked input) using a pre-trained generative, unconditioned Diffusion Model starting from a random noise (x_T) (panel (b)).

dimensionality identical to that of the original data x. In fig. 4 panel (b) discusses the approach proposed in [102] to employ pre-trained unconditional DMs to condition the generation process on filling some partial observations. The approach involves moving forward the gappy data through the Markov chain by iteratively adding noise to the observations, while simultaneously progressing backward from the noise distribution using the reverse chain learned by the DM. To incorporate the observed data in the generation process, the strategy is to repeatedly merge the forward-propagated noisy observations with the reverse-propagated noisy signal. This allows the reverse process to propagate data information within the gap and generate a correlated reconstruction. DMs have produced state-of-the-art results in image generation, see the famous example of "DALL-E 2" [103], demonstrating their ease of definition and effectiveness in training [102,104–107]. Attention [108] is another feature often implemented inside DMs architecture, which is potentially crucial for large gap-filling. Indeed, attention is meant to enhance the role of some parts of the input data while diminishing others, showing good results at handling long-range spatial relations [109]. However, DMs and attention have not yet been extensively applied in the generation and reconstruction of complex gappy flow data, but they have only been used to super-resolve smooth bi-dimensional Kolmogorov flows [110]. Therefore, the investigation of DMs and "attention" capacity to generate high-quality samples of complex flows is an ongoing field of research.

Physics-informed methods. – Leveraging the observed data and the equation of motion, physics-informed techniques exploit spatio-temporal correlations to derive accurate reconstruction of incomplete data. Kalman filters, variational approaches, and nudging are examples of advanced tools that have proven effective in enhancing initial conditions for weather forecasting problems since before ML [111–113].

Nudging is a physics-informed way to control the evolution of a flow via the continuous insertion of observed data and the addition of a penalty term, which tries to keep the flow trajectory close to that of the empirical subset [114]. Nudging has been recently applied to reconstruct high-resolution HIT flow from sparse measurements [115] and to estimate physical unknown parameters from turbulent data [116]. While physics-agnostic ML approaches are focused solely on finding patterns in data, there is growing interest in incorporating physical knowledge into ML algorithms, particularly in the field of fluid mechanics where the underlying physical laws are well understood [117]. The first objective is to impose constraints on the ML solutions to ensure that they adhere to the known physical properties, the second objective is to streamline the training by integrating relevant information directly into the network architecture or training setup. There exist three methods for incorporating physics into ML algorithms: i) observational, ii) inductive, or iii) learning biases. Observational biases may be introduced by selecting training data to ensure that a specific aspect of physics is not only present but also emphasized, *i.e.*, extreme events can be shown during training more often than the frequency at which they occur. Inductive biases embed physical constraints into the network architecture, as for example the CNNs embed invariance along the groups of symmetries possessed by typical patterns observed in images. Finally, learning biases operate in a "soft" way by adding additional terms to the loss function that penalize non-physical solutions [118], such as those that do not satisfy equations of motion, violate mass or energy conservation, and so forth. Physics-informed data-driven tools have just begun to be highlighted as particularly promising in areas as numerical weather prediction [26,119–121]. Improving data-driven and physics-informed methods synergy will undoubtedly be the focus of research in the upcoming years.

Perspectives. – Although ML techniques have been already implemented as standard tools in computer science, *fluid dynamics presents challenges that differ from those tackled in many applications of machine learning, such as image recognition and advertising*, as stated in [122]. Fluid flows necessitate precise and quantitative evaluations of the multi-scale and multi-frequency physical mechanisms that they must adhere to. On top of this, while idealized flow setups offer large datasets of high complexity, and quality, in real-life flows, one needs to deal with very sparse and noisy data. The misalignment between the idealized cases studied in the literature, and real applications opens non-trivial problems connected with the generalizability and the uncertainty quantification (UQ) of the pre-trained models [123–126]. To overcome those issues, it is highly desirable to have in the future more open-access databases, such as JHTDB (`https://turbulence.pha.jhu.edu`) and Smart-Turb (`https://smart-turb.roma2.infn.it`), and

well-defined open challenges, such as (`https://github.com/ocean-data-challenges`), that can bring different communities closer, and that can drive ML applications to go beyond theoretical exercises towards the quantitative improvement required to provide advancements in fluid mechanics. Today's challenges are connected with the need of a *quantitative AI*, driven by several critical factors such as validation, benchmarks on generalization, and UQ of ML solutions. Another crucial aspect is the *problem dimensionalization*, which involves understanding the correlation between the network's architecture, deepness, structure, and size, and the physical parameters, as Reynolds, Rayleigh, and time-to-solution, among others. As discussed, already defining an evaluation metric to quantify the solution quality is an issue in fluid mechanics that needs to be carefully designed. Answering these questions is necessary, and *interdisciplinary collaborations* between applied scientists and AI specialists are unavoidable for establishing best practices outperforming today's data assimilation techniques. Scientists are skilled at asking the right questions and they are asked to define targets that can be applied to real-world problems. AI specialists have a unique ability to "open the box" of complicated algorithms and unlock the potential of vast amounts of data.

Despite these challenges, scientific communities have not been deterred from exploring the interactions between ML and complex flows. On the contrary, the potential impact is attracting increasing attention, resulting in a convergence of challenges and new approaches that we believe are likely to continue transforming both fluid mechanics and machine learning research.

* * *

MB acknowledges Prof. Luca Biferale for useful discussion and financial support from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant Agreement No. 882340).

Data availability statement: No new data were created or analysed in this study.

REFERENCES

[1] Yeh R. A. *et al.*, *Semantic image inpainting with deep generative models*, in *Proceedings of IEEE CVPR* (IEEE) 2017, pp. 5485–5493.
[2] Ulyanov D. *et al.*, *Deep image prior*, in *Proceedings of IEEE CVPR* (IEEE) 2018, pp. 9446–9454.
[3] Bowman S. R. *et al.*, arXiv:1508.05326 (2015).
[4] Chowdhary K. R., *Natural Language Processing*, in *Fundamentals of Artificial Intelligence* (Springer, India, New Delhi) 2020, pp. 603–649.
[5] Wolf T. *et al.*, *Transformers: State-of-the-art natural language processing*, in *Proceedings of EMNLP 2020* (Association for Computational Linguistics) 2020, pp. 38–45.
[6] Jumper J. *et al.*, *Nature*, **596** (2021) 583.
[7] Biferale L. *et al.*, *Chaos*, **29** (2019) 103138.
[8] Buzzicotti M. *et al.*, *Optimal control of point-to-point navigation in turbulent time dependent flows using reinforcement learning*, in *Proceedings of AIxIA 2020* (Springer) 2021, pp. 223–234.
[9] Reddy G. *et al.*, *Nature*, **562** (2018) 236.
[10] Alageshan J. K. *et al.*, *Phys. Rev. E*, **101** (2020) 043110.
[11] Verma S. *et al.*, *Proc. Natl. Acad. Sci. U.S.A.*, **115** (2018) 5849.
[12] Calascibetta C. *et al.*, *Eur. Phys. J. E*, **46** (2023) 1.
[13] Loisy A. *et al.*, *Proc. R. Soc. A*, **478** (2022) 20220118.
[14] Loisy A., Heinonen R. A. *et al.*, *Eur. Phys. J. E*, **46** (2023) 17.
[15] Loisy A. *et al.*, arXiv:2302.00706 (2023).
[16] Bucci M. *et al.*, *Proc. R. Soc. A*, **475** (2019) 20190351.
[17] Park J. *et al.*, *J. Fluid Mech.*, **904** (2020) A24.
[18] Ren F. *et al.*, *J. Hydrodyn.*, **32** (2020) 247.
[19] Buzzicotti M. *et al.*, *Phys. Rev. Lett.*, **124** (2020) 084504.
[20] Huang X. *et al.*, *Def. Technol.*, **18** (2022) 229.
[21] Guastoni L. *et al.*, arXiv:2301.09889 (2023).
[22] Carrassi A. *et al.*, *Wiley Interdiscip. Rev. Clim. Change*, **9** (2018) e535.
[23] Reichstein M. *et al.*, *Nature*, **566** (2019) 195.
[24] Corbetta A. *et al.*, *Sci. Adv.*, **7** (2021) eaba7281.
[25] Schultz M. G. *et al.*, *Philos. Trans. R. Soc. A*, **379** (2021) 20200097.
[26] Willard J. *et al.*, *ACM Comput. Surv.*, **55** (2022) 1.
[27] Buzzicotti M. *et al.*, *Eur. Phys. J. E*, **45** (2022) 102.
[28] Bolton T. *et al.*, *J. Adv. Model Earth Syst.*, **11** (2019) 376.
[29] Park J. *et al.*, *Remote Sens.*, **11** (2019) 1366.
[30] Stock A. *et al.*, *Remote Sens.*, **12** (2020) 3313.
[31] Lou R. *et al.*, *Multimed. Syst.* (2021) 1, `https://doi.org/10.1007/s00530-020-00733-x`.
[32] Pietropolli G. *et al.*, *Gans for integration of deterministic model and observations in marine ecosystem*, in *Proceedings of 21st EPIA* (Springer) 2022, pp. 452–463.
[33] Buongiorno Nardelli B. *et al.*, *Remote Sens.*, **14** (2022) 1159.
[34] Mohan A. T. *et al.*, *J. Turbul.*, **21** (2020) 484.
[35] Woodward M. J. *et al.*, *Physics Informed Machine Learning of SPH: Machine Learning Lagrangian Turbulence* (2021), `https://openreview.net/pdf?id=bidTZROu2y`.
[36] Kim H. *et al.*, *J. Fluid Mech.*, **910** (2021) A29.
[37] Fukami K. *et al.*, *J. Fluid Mech.*, **909** (2021) A9.
[38] Buzzicotti M. *et al.*, *Phys. Rev. F*, **6** (2021) 050503.
[39] Clark Di Leoni P. *et al.*, arXiv:2210.04849 (2022).
[40] Clark Di Leoni P. *et al.*, *Eur. Phys. J. E*, **46** (2023) 16.
[41] Yousif M. Z. *et al.*, *Sci. Rep.*, **13** (2023) 2529.
[42] Nakamura T. *et al.*, *Phys. Fluids*, **33** (2021) 025116.
[43] Güemes A. *et al.*, *Phys. Fluids*, **33** (2021) 075121.
[44] Fukami K. *et al.*, *J. Fluids Eng.*, **144** (2022) 121501.
[45] Eivazi H. *et al.*, arXiv:2203.15402 (2022).
[46] Asch M. *et al.*, *Data Assimilation: Methods, Algorithms, and Applications*, Vol. **11** (SIAM) 2016.
[47] Little R. J. *et al.*, *Statistical Analysis with Missing Data*, Vol. **793** (John Wiley & Sons) 2019.
[48] Pathak D. *et al.*, *Context encoders: Feature learning*

by inpainting, in *Proceedings of IEEE CVPR* (IEEE) 2016, pp. 2536–2544.
[49] Zhu J.-Y. *et al.*, *Toward Multimodal image-to-image translation*, in *Advances in Neural Information Processing Systems*, Vol. **2017**-December (2017) pp. 466–477.
[50] Belthangady C. *et al.*, *Nat. Methods*, **16** (2019) 1215.
[51] Zavrtanik V. *et al.*, *Pattern Recognit.*, **112** (2021) 107706.
[52] Wang G. *et al.*, *IEEE Trans. Med. Imaging*, **37** (2018) 1289.
[53] Maier A. *et al.*, *Z. Med. Phys.*, **29** (2019) 86.
[54] Wang G. *et al.*, *Nat. Mach. Intell.*, **2** (2020) 737.
[55] Chai X. *et al.*, *Sci. Rep.*, **10** (2020) 3302.
[56] Wang B. *et al.*, *Geophysics*, **84** (2019) V11.
[57] Caldeira J. *et al.*, *Astron. Comput.*, **28** (2019) 100307.
[58] Moriwaki K. *et al.*, *Mon. Not. R. Astron. Soc.: Lett.*, **496** (2020) L54.
[59] Sammartino M. *et al.*, *Remote Sens.*, **12** (2020) 4123.
[60] Di J. *et al.*, *Front. Mar. Sci.*, **8** (2021) 670683.
[61] Brajard J. *et al.*, *Philos. Trans. R. Soc. A*, **379** (2021) 20200086.
[62] Shrira V. I. *et al.*, *J. Fluid Mech.*, **887** (2020) A24.
[63] Fablet R. *et al.*, *ISPRS J. Photogramm.*, **3** (2021) 295.
[64] Dong C. *et al.*, *Ocean-Land-Atmosphere Res.*, **2022** (2022) 9870950.
[65] Meneveau C. *et al.*, *Annu. Rev. Fluid Mech.*, **32** (2000) 1.
[66] Buzzicotti M. *et al.*, *J. Turbul.*, **19** (2018) 167.
[67] Biferale L. *et al.*, *Phys. Rev. Lett.*, **123** (2019) 014503.
[68] Buzzicotti M. *et al.*, *Phys. Rev. E*, **104** (2021) 015302.
[69] Duraisamy K. *et al.*, *Annu. Rev. Fluid Mech.*, **51** (2019) 357.
[70] Vinuesa R. *et al.*, *Nat. Comput. Sci.*, **2** (2022) 358.
[71] Li T. *et al.*, arXiv:2210.11921 (2022).
[72] Li T. *et al.*, arXiv:2301.07541 (2023).
[73] Storer B. A. *et al.*, *Nat. Commun.*, **13** (2022) 5314.
[74] Buzzicotti M. *et al.*, arXiv:2106.04157 (2021).
[75] Pujol M.-I. *et al.*, *Ocean Sci.*, **12** (2016) 1067.
[76] Ballarotta M. *et al.*, *Ocean Sci.*, **15** (2019) 1091.
[77] Sirovich L. *et al.*, *J. Opt. Soc. Am. A*, **4** (1987) 519.
[78] Fukunaga K., *Introduction to Statistical Pattern Recognition* (Elsevier) 2013.
[79] Hayase T., *Fluid Dyn. Res.*, **47** (2015) 051201.
[80] Kondrashov D. *et al.*, *Nonlinear Process. Geophys.*, **13** (2006) 151.
[81] Everson R. *et al.*, *J. Opt. Soc. Am. A*, **12** (1995) 1657.
[82] Maurel S. *et al.*, *Flow Turbul. Combust.*, **67** (2001) 125.
[83] Gunes H. *et al.*, *J. Comput. Phys.*, **212** (2006) 358.
[84] Guastoni L. *et al.*, *J. Fluid Mech.*, **928** (2021) A27.
[85] Hong Y. *et al.*, *ACM Comput. Surv.*, **52** (2019) 1.
[86] Bond-Taylor S. *et al.*, *IEEE Trans. Pattern Anal. Mach. Intell.*, **44** (2022) 7327.
[87] Gui J. *et al.*, *IEEE Trans. Knowl. Data Eng.*, **35** (2023) 3313.
[88] O'Shea K. *et al.*, arXiv:1511.08458 (2015).
[89] Li Z. *et al.*, *IEEE Trans. Neural Netw. Learn. Syst.*, **33** (2022) 6999.
[90] Eivazi H. *et al.*, *Expert Syst. Appl.*, **202** (2022) 117038.
[91] Goodfellow I. *et al.*, *Adv. Neural Inf. Process. Syst.*, **27** (2014) 2672.
[92] Wang K. *et al.*, *IEEE/CAA J. Autom. Sin.*, **4** (2017) 588.
[93] Heusel M. *et al.*, *GANs trained by a two time-scale update rule converge to a local Nash equilibrium*, in *Advances in Neural Information Processing Systems*, Vol. **2017**-December (2017) pp. 6627–6638.
[94] Hofbauer *et al.*, *Evolutionary Games and Population Dynamics* (Cambridge University Press) 1998.
[95] Borra F. *et al.*, *Phys. Rev. F*, **7** (2022) 023103.
[96] Deng Z. *et al.*, *Phys. Fluids*, **31** (2019) 125111.
[97] Subramaniam A. *et al.*, arXiv:2003.01907 (2020).
[98] Kingma D. P. *et al.*, arXiv:1312.6114 (2013).
[99] Salimans T. *et al.*, *Markov chain monte carlo and variational inference: Bridging the gap*, in *Proceedings of ICML* (PMLR) 2015, pp. 1218–1226.
[100] Doersch C., arXiv:1606.05908 (2016).
[101] Sohl-Dickstein J. *et al.*, *Deep unsupervised learning using nonequilibrium thermodynamics*, in *Proceedings of ICML* (PMLR) 2015, pp. 2256–2265.
[102] Lugmayr A. *et al.*, *Repaint: Inpainting using denoising diffusion probabilistic models*, in *Proceedings of IEEE CVPR* (IEEE) 2022, pp. 11461–11471.
[103] Saharia C. *et al.*, arXiv:2205.11487 (2022).
[104] Ho J. *et al.*, *Adv. Neural Inf. Process. Syst.*, **33** (2020) 6840.
[105] Nichol A. Q. *et al.*, *Improved denoising diffusion probabilistic models*, in *Proceedings of ICML* (PMLR) 2021, pp. 8162–8171.
[106] Dhariwal P. *et al.*, *Adv. Neural Inf. Process. Syst.*, **34** (2021) 8780.
[107] Rombach R. *et al.*, *High-resolution image synthesis with latent diffusion models*, in *Proceedings of IEEE CVPR* (IEEE) 2022, pp. 10684–10695.
[108] Vaswani A. *et al.*, *Adv. Neural Inf. Process. Syst.*, **30** (2017).
[109] Wang X. *et al.*, *Non-local neural networks*, in *Proceedings of IEEE CVPR* (IEEE) 2018, pp. 7794–7803.
[110] Shu D. *et al.*, *J. Comput. Phys.*, **478** (2023) 111972.
[111] Kalnay E., *Atmospheric Modeling, Data Assimilation and Predictability* (Cambridge University Press) 2003.
[112] Carrassi A. *et al.*, *Wiley Interdiscip. Rev. Clim. Change*, **9** (2018) e535.
[113] Lakshmivarahan S. and Lewis J. M., *Nudging methods: A critical overview*, in *Data Assimilation for Atmospheric, Oceanic and Hydrologic Applications*, Vol. **II** (Springer) 2013, pp. 27–57.
[114] Hoke J. E. *et al.*, *Mon. Weather Rev.*, **104** (1976) 1551.
[115] Clark Di Leoni P. *et al.*, *Phys. Rev. X*, **10** (2020) 011023.
[116] Buzzicotti M. *et al.*, *Phys. Fluids*, **32** (2020) 125116.
[117] Karniadakis G. E. *et al.*, *Nat. Rev. Phys.*, **3** (2021) 422.
[118] Raissi M. *et al.*, *J. Comput. Phys.*, **378** (2019) 686.
[119] Zhao W. L. *et al.*, *Geophys. Res. Lett.*, **46** (2019) 14496.
[120] Alber M. *et al.*, *NPJ Digit. Med.*, **2** (2019) 115.
[121] Kashinath K. *et al.*, *Philos. Trans. R. Soc. A*, **379** (2021) 20200093.
[122] Brunton S. L. *et al.*, *Annu. Rev. Fluid Mech.*, **52** (2020) 477.
[123] Bucci M. *et al.*, *Eur. Phys. J. E*, **46** (2023) 12.
[124] Abdar M. *et al.*, *Inf. Fusion*, **76** (2021) 243.
[125] Hüllermeier E. *et al.*, *Mach. Learn.*, **110** (2021) 457.
[126] Barth A. *et al.*, *Geosci. Model Dev.*, **13** (2020) 1609.