Physics-Driven Probabilistic Deep Learning for the Inversion of Physical Models With Application to Phenological Parameter Retrieval From Satellite Times Series

Yoël Zérah, Silvia Valero, Jordi Inglada

Physics-driven probabilistic deep learning for the inversion of physical models with application to phenological parameter retrieval from satellite times series

Yoël Zérah, Silvia Valero, Jordi Inglada

***Abstract*—Recent Sentinel satellite constellations and deep learning methods offer great possibilities for estimating the states and dynamics of physical parameters on a global scale. Such parameters and their corresponding uncertainties can be retrieved by machine learning methods solving probabilistic inverse problems. Nevertheless, the scarcity of reference data to train supervised methodologies is a well-known constraint for remote sensing applications. To address such limitations, this work presents a new generic physics-guided probabilistic deep learning methodology to invert physical models. The presented methodology proposes a new strategy to combine probabilistic deep learning methods and physical models avoiding simulation-driven machine learning. The inverse problem is addressed through a Bayesian inference framework by proposing a new physically-constrained self-supervised representation learning methodology. To show the interest of the proposed strategy, the methodology is applied to the retrieval of phenological parameters from NDVI time series. As a result, the probability distributions of the intrinsic phenological model parameters are inferred. The feasibility of the method is evaluated on both simulated and real Sentinel-2 data and compared with different standard algorithms. Promising results show satisfactory accuracy predictions and low inference times for real applications.**

***Index Terms*—Generative Models, Autoencoders, Satellite Image Time Series, Self-Supervised Representation Learning, Bayesian physics-guided learning, Inverse problems, Phenology Monitoring, Large Scale.**

I. Introduction

NOWADAYS, vast amounts of data are acquired by satellite-borne sensors for Earth Observation. In the last decade, the Sentinel-2 (S2) satellites of EU's Copernicus program have been acquiring optical Satellite Image Time Series (SITS), with high spatial, spectral and temporal resolutions. These data enable large scale applications involving Earth monitoring with stunning precision. In particular, S2 images provide a valuable source of information to retrieve parameters which have physical meaning related to properties of land surface.

This work was supported by the Natural Intelligence Toulouse Institute (ANITI) from the Université Fédérale Toulouse Midi-Pyrenées under Grant Agreement ANITI ANR-19-P3IA-0004, by the ANR-JCJC DeepChange project under Grant Agreement 20-CE23-0003, and by the Centre National d'Études Spatiales (CNES) under Grant Agreement n° 51/19560.

Y. Zérah, S. Valero, J. Inglada are with CESBIO, Université de Toulouse, CNES/CNRS/INRAe/IRD/UPS, 31000 Toulouse, France (e-mail: yoel.zerah@univ-toulouse.fr, silvia.valero@cesbio.cnes.fr, jordi.inglada@cesbio.cnes.fr).

Vegetation parameters estimated through remote-sensing are essential to understand the ecosystems on earth. For instance, biophysical parameters are highlighted as essential climate variables (ECVs) supporting numerous applications in agriculture, forestry and climate change. Phenological indicators have also a great relevance for studying plant diversity, vegetation structure and ecosystem change. The common agricultural policy (CAP) supporting farmers has recently highlighted the important role of crop phenology directly controlling productivity [1].

Physical models describing the interaction between the observed satellite data and the parameter under observation play a key role for parameter estimation. The inversion of physical models is usually proposed to infer the values of the parameters characterizing the physical equations [2], [3]. For instance, the well-known double-logistic phenological model is considered for phenological parameter retrieval. The inversion of this model is traditionally applied on Normalized Difference Vegetation Index (NDVI) times series which quantify vegetation dynamics [4]–[7].

Inversion methods in remote sensing have to face two major challenges: (i) the diversity and complexity of landscape to be studied and (ii) the scarcity of labeled data — although this issue also plagues many remote sensing applications, triggering the need to develop unsupervised methods [8]. Traditionally, the lack of human annotations is mitigated by generating simulations from the physical model considering a high number of combinations of input parameter values.

Inversion methods for parameter retrieval have exploited in different ways the input-output simulation data pairs. For methods based on look-up tables (LUT), the inversion problem is reduced to the identification of the simulation set that resembles most closely the measured one. As large tables are required to simulate Earth observation scenarios, these spectrum-matching techniques are highly inefficient and extremely slow [9].

Recently, some methods have proposed the combination of traditional physics-based modeling approaches with state-of-the-art machine learning techniques [10]–[12]. To solve the inverse problem, these hybrid methods propose to train statistical regression algorithms with physics-based simulations. One of the main advantages of using machine learning is that these supervised models can find structure and patterns in complex satellite data where physical processes are not fully

understood. Therefore, informed machine learning methods offer the combination of trainability and generalization of machine learning while respecting the physical equations [13].

Despite being a very promising solution, these simulation-driven regression methods involve important limitations for large scale parameter retrieval applications. The main drawback is that simulations can not be always realistic and the reality gap with the the real-world behavior might be large. In this situation, machine learning models are not able to capture relationships for unavailable training data situations, and thus cannot generalize to out-of-sample scenarios. To overcome these limitations, these simulated assisted methods require large fine-tuning process to simulate training data sets that can generalize (or be transferred) [14]. Another drawback of these hybrid approaches is that simulations can be hindered by uncertainties of real satellite data or unaccounted physical phenomena which can introduce additional bias.

Probabilistic methods, such as Monte Carlo approaches, provide a potential model inversion solution to avoid the generation of synthetic training data scenarios. In remote sensing, good performances are obtained by sampling intensive approaches such as the Markov Chain Monte Carlo (MCMC) [15]. Besides, these methodologies provide a reliable approach to quantify the uncertainty of the retrieved parameters. Unfortunately, these strategies can not be applied on large scale inference problems because of its high computational cost.

To overcome the above mentioned drawbacks, this work presents a new generic physics-guided probabilistic deep learning methodology to invert physical models. The inverse problem is addressed through a Bayesian framework by proposing a new physically-constrained self-supervised representation learning methodology. The proposed approach is based on Variational Autoencoders (VAE), which combine Bayesian theory with deep learning [16], [17]. The main idea is to consider that input parameters of a physical model are a generative representation of data. Assuming that, the proposed self-supervised methodology learns to infer latent variables corresponding to the probability distributions of the intrinsic physical model parameters, without relying on scarce reference data or simulated data-sets.

To corroborate the potential of the proposed method, the inversion of the above mentioned phenological vegetation model is presented as an example. The presented *pheno-VAE* strategy shows how physical priors can be incorporated into the self-supervised learning process. As a result, probabilistic latent distributions semantically bound to input model parameters can be retrieved.

The remainder of this article is organized as follows. Section II introduces how self-supervised representation learning based on VAE can be used to solve inverse problems. Section III presents how physical priors can be incorporated into the learning process to estimate physical model parameters as latent variables. Section IV describes *pheno-VAE* where the proposed methodology is applied to invert a classical vegetation phenological model on satellite optical time series. Finally, experimental results shown in Section V corroborate the interest of the presented model inversion strategy.

II. Related works

A. Representation learning with Variational Autoencoders

Autoencoders (AE) are self-supervised neural networks that learn low dimensional representations from unlabeled data. An encoder reduces the dimension of the input data into deterministic latent variables that are used by the decoder to reconstruct the input data. Both the encoder and the decoder are neural networks that are trained simultaneously to optimize the compression of the input data. The loss is usually a mean squared error (MSE) of the reconstruction. VAE (see Fig. 1) embed the representation in the latent space,

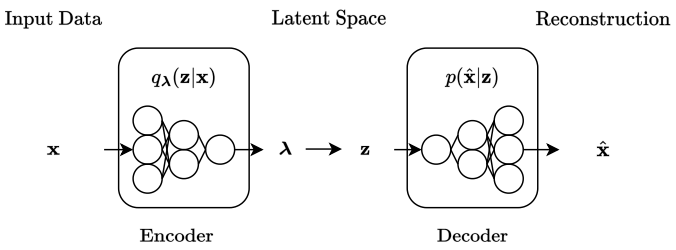


Fig. 1. Classical Variational Autoencoder

as random variables [16]. Specifically, the encoder outputs the parameters $\boldsymbol{\lambda}$ of a so-called *variational* distribution. The variational distribution belongs to a parametric distribution family, usually a Gaussian distribution. The VAE latent space being a distribution, it fosters regular and continuous representations, making it more suitable to represent high level features. Then the realizations $\mathbf{z}$ of this distribution are taken as input to the decoder. The decoder's output are also distribution parameters, from which reconstructions $\hat{\mathbf{x}}$ are sampled. This is sometimes called *ancestral sampling* [18].

A well-known problem of VAE is the collapse of latent variables [19], [20]. When the decoder is a too powerful generative model, it is able to reconstruct the input while ignoring a subset of the latent variables. In this case, latent variables can become redundant, and their distributions collapse. Another challenge of representation learning with VAE is the model identification issue related to the classical standard Gaussian prior. Despite the existence of many distribution families, most works only consider Gaussian latent spaces. This can limit the ability of VAE to infer meaningful representations. In fact, the latent distributions learned by traditional VAE architectures aren't easily interpretable because associated latent variables are the generative factors of an unknown generative model.

B. Solving inverse problems with representation learning

Hybrid methods combining physics with deep learning offer a new approach for solving remote sensing problems [13]. The integration of physical priors into representation learning methods is a promising solution to discover semantic latent variables tied to the parameters of a generative physical model [21], [22].

Theoretically, three types of priors can be introduced to guide learning toward physically consistent predictions [23] (see Fig. 2): "observational biases", "inductive biases" and "learning biases". Observational biases are brought through

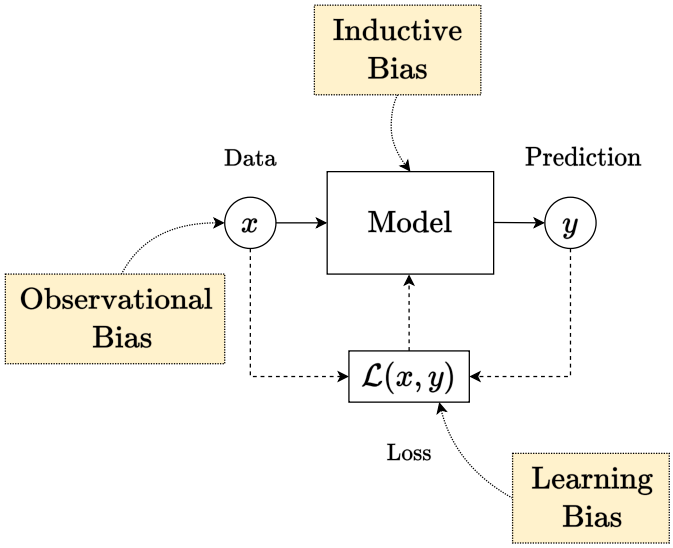


Fig. 2. Incorporation of physical priors in machine learning models.

the choice of data that capture the physical properties of interest. Inductive biases are incorporated by the tailoring of models so that predictions are guaranteed to follow specified physical behaviors. Learning biases are enforced through the choice of loss functions. Several recent methods based on generative models integrate physics with learning and induction biases by specifically tweaking the generative processes [24], [25]. In [26], spectral unmixing, which is a common remote sensing inverse problem, is performed by enforcing specific constraints between the latent variables of a VAE. Some strategies propose the incorporation of inductive bias by integrating parametric physical models into representation learning methodologies. These methods, which are based on encoder-decoder architectures, consider physical equations as a known generative model. Considering that, these strategies replace the classical neural network decoder by a user-defined model. As a result, the learned variables are semantically tied to the model parameters and the encoder is trained to approximate the inverse model. This solution performs parameter inference with a simple forward pass of the input data [27][28]. From the representation learning perspective, variables predicted from an inverted model can be thought of as an interpretable representation.

As physical models are not perfect, some works such as [27] propose to infer representations that are partially interpretable. For these strategies latent space has: (i) an interpretable part bound to a user-defined decoder and (ii) a non interpretable part bound to a neural network decoder.

The above described hybrid strategy is proposed in this work to solve remote sensing problems. Beside proposing the integration of a physical decoder, different strategies to incorporate inductive and learning biases from physical data are presented. To retrieve accurate uncertainties derived from probabilistic latent variables, a novel training procedure based on the sampling of the latent distribution is also proposed.

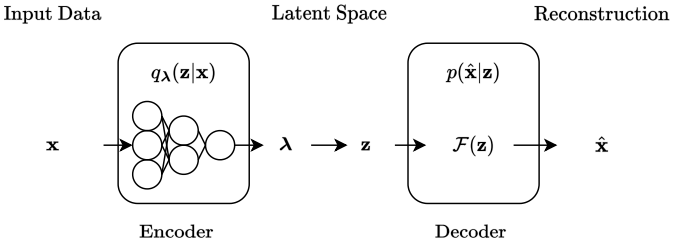


Fig. 3. VAE with user-defined decoder

III. Methodology

The theoretical basis of VAE is firstly introduced through the perspective of variational inference. Secondly, different methodological contributions are presented to incorporate physical priors through inductive and learning biases: (i) a new Monte-Carlo reconstruction loss strategy for the incorporation of physical models in VAE decoders, (ii) the possibility and benefit of using variational distributions other than Gaussian to better model physical quantities, (iii) the incorporation of physical priors by imposing complementary relationship constraints on latent distributions.

A. Amortized variational inference with VAE

Let there be a probabilistic model that has observations $\mathbf{x}$ and latent variables $\mathbf{z}$, with its joint density:

$$p(\mathbf{x}, \mathbf{z}) = p(\mathbf{x}|\mathbf{z})p(\mathbf{z}) \tag{1}$$

with $p(\mathbf{z})$ the *prior* over the latent distribution and $p(\mathbf{x}|\mathbf{z})$ the *likelihood*. It should be noted that $p(\mathbf{x}, \mathbf{z})$ is a *generative model* of observations from latent variables, and $\mathbf{z}$ is then a *generative factor*, and a representation of $\mathbf{x}$. Computing the posterior $p(\mathbf{z}|\mathbf{x})$ is known as the inference problem. Although Bayes theorem,

$$p(\mathbf{z}|\mathbf{x}) = \frac{p(\mathbf{x}|\mathbf{z})p(\mathbf{z})}{p(\mathbf{x})} = \frac{p(\mathbf{x}|\mathbf{z})p(\mathbf{z})}{\int p(\mathbf{x}, \mathbf{z})d\mathbf{z}}, \tag{2}$$

defines a rigorous mathematical formulation for any inference problem, it is not directly applicable. This is because $\int p(\mathbf{x}, \mathbf{z})d\mathbf{z}$ can become intractable due to the large dimensionality of $\mathbf{z}$. To overcome this issue, instead of calculating the exact posterior, approximation methods are commonly used. In particular, variational inference methods approximate the posterior with a so-called *variational distribution* $q_{\boldsymbol{\lambda}}(\mathbf{z}|\mathbf{x})$, that is restricted to belong to a $\boldsymbol{\lambda}$-parameterized distribution family $\mathcal{Q}_{\boldsymbol{\lambda}}$.

To ensure that $q_{\boldsymbol{\lambda}}(\mathbf{z}|\mathbf{x})$ is the best approximation of the posterior among $\mathcal{Q}_{\boldsymbol{\lambda}}$, inference methods minimize the Kullback-Leibler (KL) divergence between the posterior and its approximation:

$$q^*_{\boldsymbol{\lambda}}(z) = \underset{q_{\boldsymbol{\lambda}} \in \mathcal{Q}_{\boldsymbol{\lambda}}}{\arg\min}\ \mathbb{KL}\left(q_{\boldsymbol{\lambda}}(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z}|\mathbf{x})\right). \tag{3}$$

The KL-divergence is also untractable here because of the evidence term, $\log p(\mathbf{x}) = \log \int p(\mathbf{x}, \mathbf{z})d\mathbf{z}$.

$$\begin{aligned} \mathbb{KL}\left(q_{\boldsymbol{\lambda}}(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z}|\mathbf{x})\right) = \mathbb{E}\left[\log q_{\boldsymbol{\lambda}}(\mathbf{z}|\mathbf{x})\right] \\ - \mathbb{E}\left[\log p(\mathbf{x}, \mathbf{z})\right] \\ + \log p(\mathbf{x}) \end{aligned} \tag{4}$$

The optimization problem can be solved by using the ELBO denoted in (5), by considering a prior distribution $p(\mathbf{z})$ over the variational distribution.

$$\text{ELBO}\left(q_{\boldsymbol{\lambda}}\right) = \mathbb{E}\left[\log p(\mathbf{x}|\mathbf{z})\right] - \mathbb{KL}\left(q_{\boldsymbol{\lambda}}(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z})\right) \tag{5}$$

Because the evidence is constant with respect to $\boldsymbol{\lambda}$, maximizing the ELBO leads to minimizing the KL divergence term in (3).

In VAE, the likelihood $p(\mathbf{x}|\mathbf{z},\theta)$ is embedded in the decoder, and the posterior distribution $q_{\boldsymbol{\lambda}}(\mathbf{z}|\mathbf{x},\phi)$ in the encoder, with θ and ϕ the respective networks' parameters. The encoder infers the variational parameters $\boldsymbol{\lambda}$. The variational distribution is typically chosen to be Gaussian: $q_{\boldsymbol{\lambda}}(\mathbf{z}|\mathbf{x},\phi) = \mathcal{N}(\mathbf{z}|\boldsymbol{\lambda})$ with $\boldsymbol{\lambda} = [\boldsymbol{\mu}_{\mathbf{z}}(\mathbf{x},\phi), \boldsymbol{\Sigma}_{\mathbf{z}}(\mathbf{x},\phi)]$, with $\boldsymbol{\mu}_{\mathbf{z}}$ and $\boldsymbol{\Sigma}_{\mathbf{z}}$ being the mean vector and the covariance matrix of the latent variables. This choice enables the explicit computation of the KL loss term with a standard Gaussian prior, and a differentiable sampling strategy[1] using the reparameterization trick (see (6)). In practice, $\boldsymbol{\Sigma}_{\mathbf{z}}$ is assumed to be a diagonal matrix, because it prevents having to ensure definite positiveness and it reduces the number of inferred latent parameters.

$$\mathbf{z} = \boldsymbol{\mu}_{\mathbf{z}} + \boldsymbol{\Sigma}_{\mathbf{z}}^{1/2}\boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}\left(\mathbf{0},\mathbf{I}\right) \;\Rightarrow\; \mathbf{z} \sim \mathcal{N}\left(\boldsymbol{\mu}_{\mathbf{z}}, \boldsymbol{\Sigma}_{\mathbf{z}}\right) \tag{6}$$

The decoder infers the parameters of the distribution of the reconstructions selected among a chosen parametric distribution family — although this aspect is often overlooked in the literature. In this work, the distribution of the decoder is chosen as Gaussian: $p(\hat{\mathbf{x}}|\mathbf{z},\theta) = \mathcal{N}(\hat{\mathbf{x}}|\boldsymbol{\mu}_{\hat{\mathbf{x}}}(\mathbf{z},\theta), \boldsymbol{\Sigma}_{\hat{\mathbf{x}}}(\mathbf{z},\theta))$, with $\boldsymbol{\mu}_{\hat{\mathbf{x}}}$ and $\boldsymbol{\Sigma}_{\hat{\mathbf{x}}}$ the corresponding mean vector and covariance matrix. The covariance matrix $\boldsymbol{\Sigma}_{\hat{\mathbf{x}}}$ is commonly set as a hyper-parameter (often set to identity matrix). It can also be considered a trainable parameter or estimated from the input's distribution [29].

Traditionally, the negative ELBO (5) is the loss function minimized during the VAE training process. It has two terms : $\mathcal{L} = \mathcal{L}_{rec} + \mathcal{L}_{KL}$. The reconstruction term $\mathcal{L}_{rec} = -\mathbb{E}\left[\log p(\mathbf{x}|\mathbf{z})\right]$ is the expectation of the Negative Log Likelihood (NLL). It forces decoded samples to match the initial input data. $\mathcal{L}_{rec}$ can be approximated as the average NLL over a number L of Monte-Carlo samples of the latent distribution (see (7)). However in practice with a batch size large enough, $\mathbf{z}$ is typically only sampled once per iteration [16].

$$\mathbb{E}\left[\log p(\mathbf{x}|\mathbf{z})\right] \approx \frac{1}{L}\sum_{i=1}^{L} \log p(\mathbf{x}|\mathbf{z}^{(i)}) \tag{7}$$

Typically, this reconstruction loss term corresponds to the MSE assuming classical a unit-variance Gaussian decoder distribution. Unfortunately, this assumption tends to over-regularized VAE not allowing accurate uncertainty predictions.

The second term of the ELBO $\mathcal{L}_{KL} = \mathbb{KL}\left(q_{\boldsymbol{\lambda}}(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z})\right)$, is a regularization term that penalizes the mismatch of the variational distributions to the prior $p(\mathbf{z})$. This term has a closed form with Gaussian latent spaces and the usual prior $\mathcal{N}(\mathbf{0},\mathbf{I})$.

[1] with respect to variational parameters

B. Monte Carlo reconstruction loss for deterministic decoders

The informed deep learning methodology presented in this work proposes the use of physical-based decoders $\mathcal{F}$ as shown in Fig. 3. It implies that decoder outputs can no longer be distribution parameters since only samples from $p(\mathbf{x}|\mathbf{z})$ are available for reconstruction loss computation. Therefore, we propose to approximate $p(\mathbf{x}|\mathbf{z})$ as a Gaussian distribution by estimating its means and covariance parameters (Eq.(8) and Eq.(9)). For each sample, both parameters are estimated by considering several reconstructions obtained by applying a Monte-Carlo sampling strategy on the corresponding latent distributions. The difference between a classical Gaussian decoder and our proposed approach is depicted in Fig. 4).

$$\boldsymbol{\mu}_{\hat{\mathbf{x}}}(\mathbf{z}) \approx \frac{1}{K}\sum_{i=1}^{K} \mathcal{F}(\mathbf{z}^{(i)}) \tag{8}$$

$$\boldsymbol{\Sigma}_{\hat{\mathbf{x}}}(\mathbf{z}) \approx \frac{1}{K-1}\sum_{i=1}^{K} \left(\mathcal{F}(\mathbf{z}^{(i)}) - \boldsymbol{\mu}_{\hat{\mathbf{x}}}\right)\left(\mathcal{F}(\mathbf{z}^{(i)}) - \boldsymbol{\mu}_{\hat{\mathbf{x}}}\right)^{\top} \tag{9}$$

The Monte-Carlo sampling of latent space brings up a new hyper-parameter K: the number of latent samples drawn from the latent distribution inferred from each input sample $\mathbf{x}$. The choice of K is a trade-off between accuracy of $\boldsymbol{\mu}_{\hat{\mathbf{x}}}$ and $\boldsymbol{\Sigma}_{\hat{\mathbf{x}}}$, and training time, because latent distribution sample requires a forward pass through the decoder. Considering this, the reconstruction loss term for the proposed Gaussian decoder can be computed as

$$\mathcal{L}_{rec}(\mathbf{x}) = \frac{1}{2}\left[\left(\mathbf{x} - \boldsymbol{\mu}_{\hat{\mathbf{x}}}\right)^{\top} \boldsymbol{\Sigma}_{\hat{\mathbf{x}}}^{-1}\left(\mathbf{x} - \boldsymbol{\mu}_{\hat{\mathbf{x}}}\right) + \ln\left(|\boldsymbol{\Sigma}_{\hat{\mathbf{x}}}|\right)\right] \tag{10}$$

$\Sigma_{\hat{\mathbf{x}}}$ is approximated as a diagonal covariance matrix by assuming the independence of reconstruction components. Discarding the assumption of the covariance matrix diagonality could improve reconstruction quality, and add structure to residuals [30]. However, covariance matrix inversion and determinant computation would become prohibitively expensive for any large dimensional data.

The Gaussian NLL encourages both the reconstruction error of each sample to be small, and the reconstruction variance to model uncertainty, even if the distribution of reconstructions is not Gaussian. If the error isn't small, the variance can be increased to still minimize the loss (e.g. when the error cannot be minimized, uncertainty is increased). The $\ln\left(|\Sigma_{\hat{x}}|\right)$ term prevents the variance from arbitrarily increasing as a trivial way of minimizing the loss.

C. The variational distribution as an inductive bias

The use of a physical-based decoder implies that latent variables are tied to physical measurements. Therefore, knowledge about the probability distribution characterizing these measurements can be used to discard the classical prior and posterior Gaussian assumption. We advocate for choosing a variational distribution family that matches assumptions about each semantic latent variables, and a prior that accounts for knowledge about the data-set.

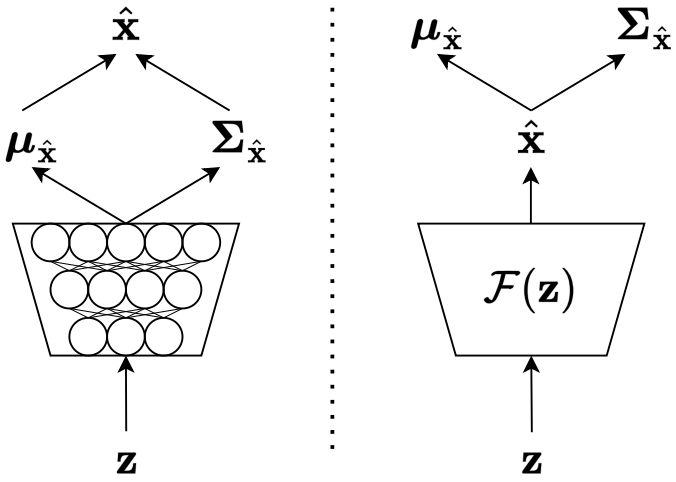


Fig. 4. Predicting parameters of a Gaussian decoder. Left: original Gaussian decoder, with both the mean and the covariance being output by a trained neural network. Right: a non-trainable decoder, where the mean and variance are estimated from a set of K reconstructed samples $\hat{\mathbf{x}}$ obtained by the Monte-Carlo sampling strategy.

The choice of the variational distribution is limited to distributions that can be sampled in a differentiable way, so that gradients can be propagated through. Three different sampling techniques can be considered to enable various distribution choices [16]:

1) A reparameterization trick to sample *location-scale family* distributions [31], such as the usual Gaussian distribution (see (6)).
2) The composition of random variables by non-linear functions enables to transform "elementary" distributions into others. For instance, log-normal, logit-normal, Dirichlet, exponential distribution samples can be generated respectively by composing Gaussian with logarithm, Gaussian with sigmoid, Gaussian with softmax [32] and uniform with logarithm.
3) The inverse transform sampling method described in (11) can be used to sample any continuous random variable $z \sim \mathcal{A}$. This technique can be used since its inverse cumulative distribution function (ICDF) $F_{\mathcal{A}}^{-1}$ is is differentiable almost everywhere. It entails sampling u from $\mathcal{U}(0,1)$ (the uniform distribution), and then calculating the desired z as:.

$$z = F_{\mathcal{A}}^{-1}(u), \quad u \sim \mathcal{U}(0,1) \quad \Rightarrow \quad z \sim \mathcal{A} \tag{11}$$

In practice, the gradient of the ICDF computed during training may diverge. In intervals of zero density, the CDF is constant at $y = c$ and its reciprocal has infinite derivative at $x = c$. Therefore uniform sampling of u must be done inside an interval I where the CDF is strictly monotonous. In fact, due to the numerical precision ϵ, the interval I has to be restricted even further (see (12)).

$$I = F_{\mathcal{A}}(X), \; X = \{x \in [0,1] \text{ s.t. } dF_{\mathcal{A}}(x) \geq \epsilon\} \tag{12}$$

As mentioned above, learned latent variables correspond to physical measures being them typically bounded. If parameters to be estimated are known to belong to a certain bounded interval, their corresponding variational distributions should have closed-support. Physical models can be mathematically defined even with out-of-bounds parameters, but samples generated with these parameters would not be realistic. Such reconstructions could still minimize the reconstruction loss, and hamper training while the encoder learns to infer wrong models parameters. This can be especially detrimental when some training samples are not well described by the physical model. The VAE would then bend the model instead of increasing latent uncertainty.

The above-described sampling techniques can be used to sample bounded distributions. This can be achieved by composing unbounded distribution samples, such as Gaussian samples drawn from the reparameterization trick, with sigmoid functions[2] (logistic[3], hyperbolic tangent, arc-tangent, etc...). However with this method, the resulting distributions are distorted, asymmetric near the support bounds and may even become bimodal. To avoid such limitations, the inverse transform method enables the sampling of closed support distributions such as raised cosine distributions, Kumaraswamy distributions, etc.

The bounds of the distributions can be inferred by the encoder, or set by the user. In both cases, it may be convenient to sample bounded distributions on the interval $[0,1]$ and then perform affine scaling to the desired $[a,b]$ interval (see (13)).

$$z \in [0,1] \Rightarrow (b-a)z + a \in [a,b] \tag{13}$$

While the variational distribution should be chosen with physical variables meaning in mind, it has to be paired with a prior distribution that enables computation of the KL loss term. It may unfortunately be more complicated to find a meaningful prior whose KL divergence with the variational distribution admits a closed-form expression.

D. Incorporation of order constraints into latent distributions

The presented sampling methods assume the independence of latent variables which could not be true when imposing interpretability in the encoded space. Physical variables of a model may not be independent, and correlations and statistical dependence can be typically observed between variables. For instance, physical models associated to satellite time series usually have input parameters associated to time, therefore order relationships can be established. Different strategies are proposed here to introduce dependence between latent variables, while still performing independent sampling as is done with classical VAE. By considering ancestral sampling [18], we propose to constraint latent variables by an order relation. Furthermore, to prevent the training from converging to model parameters that are not physically plausible, additional constraints are enforced.

Let there be n latent variables z_i, on intervals $[a_i, b_i]$ that must be ordered as follows: $z_i < z_{i+1}, \quad \forall i \in [\![1, n-1]\!]$. Two complementary situations can arise:

[2]More generally, composing unbounded samples with a monotonic, smooth enough, bounded function.

[3]The composition of Gaussian distribution with logistic function is the *logit-normal distribution.*

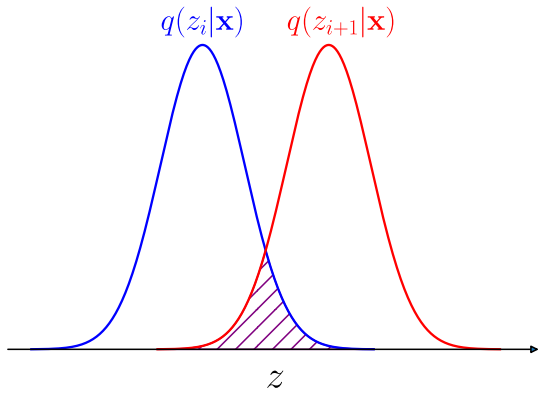


Fig. 5. Marginal densities $q(z_i|\mathbf{x})$ and $q(z_{i+1}|\mathbf{x})$ of latent variables z_i and z_{i+1} with intersecting support. If sampled independently, there is a non-zero probability that $z_i > z_{i+1}$.

(i) $\forall i,\ [a_i, b_i] \cap [a_{i+1}, b_{i+1}] = \emptyset$. There is no intersection between the support of each two consecutive latent variable distribution.
(ii) $\exists i$ such that $[a_i, b_i] \cap [a_{i+1}, b_{i+1}] \neq \emptyset$. There is some intersection between the support of two consecutive latent distributions.

In the first case, the independent sampling of each z_i will always yield ordered results. In the second case, there is a non-zero probability of sampling disordered samples z_i and z_{i+1} from two consecutive latent distributions $q(z_i|\mathbf{x})$ and $q(z_{i+1}|\mathbf{x})$ (see Fig. 5). To ensure that latent samples are always ordered in this second situation, we identify the three following strategies:

1) Penalizing out-of-order latent samples: Enforcing ordered constraints by just penalizing latent samples that are out of order would decrease the latent distributions widths and prevent latent distributions from being too close. As a result, the encoder would arbitrarily infer disjoint marginal distributions. This solution is not applicable in general, because it introduces an inductive prior of distribution disjointedness that is not necessarily assumed by the physical-based decoder. Furthermore it would also hamper training by introducing noise into the loss.

2) Inferring the distribution of the difference between two variables: The second way of ensuring the order of samples is to infer positive support distributions of the difference $\Delta_{z_{i+1}}$ between each pair of consecutive variables (z_{i+1}, z_i) in the ordered sequence (see (14)).

$$z_{i+1} = z_i + \Delta_{z_{i+1}}, \quad \forall i \in [\![1, n-1]\!] \tag{14}$$

However this method increases the variance of the summed latent variables. In fact, the density of the sum of random variables is the convolution of the densities of these variables, and the convolution of two densities results in a wider density.

3) Inferring the distribution of the maximum of two variables: To overcome previous methods shortcomings, we propose to use the distribution of the maximum of two consecutive variables (z_{i+1}, z_i) as the distribution of the greater variable z_{i+1}.

To perform that, for consecutive latent distributions, it is necessary to ensure that samples are ordered, and that the expectations of the distributions are ordered. The former is achieved with the rectification of latent samples, while the latter is attained with the rectification of the variational parameters and with the use an additional loss term on the variational parameters. The sampling procedure of ordered latent variables is illustrated in Fig. 6.

To *rectify* latent samples, the maximum value between a sample of consecutive variables (z_i, z_{i+1}) is attributed to the the greater variable z_{i+1} (see (15)). If the samples are ordered beforehand, the rectification doesn't change the value of the greater variable. If samples were disordered, this sets the value of the greater variable as equal to that of the lower variable z_i. The resulting rectified samples $\underline{z}_{i+1}$ are then used instead of z_{i+1} by the user-defined decoder.

$$\underline{z}_{i+1} = \max\left(z_{i+1}, z_i\right), \quad \forall i \in [\![1, n-1]\!] \tag{15}$$

The distribution of each z_i is then the distribution of the maximum of all previous consecutive variables $z_i = \max_{j \leq i}(z_j), \quad \forall i \in [\![1, n-1]\!]$. The density (PDF) and cumulative distribution function (CDF) of rectified latent variables are available (see appendix D) if the PDF and CDF of all marginal latent distributions are available (marginal distributions can even be from different distribution families). Sample rectification is effective when distributions of consecutive latent variables overlap.

Since the rectification step takes place after the variational parameters inference, the model may rely solely on the rectification step to produce ordered latent variables. When the expectations of two consecutive latent distributions $q(z_i|\mathbf{x})$ and $q(z_{i+1}|\mathbf{x})$ are disordered, the rectification step will mostly make consecutive latent samples identical. The encoder might converge sub-optimally and even though latent samples would technically be ordered, they would never be the right value.

To mitigate this, the expectation of consecutive latent distributions must be ordered as well. The two additional proposed techniques aim at ensuring that the encoder outputs variational parameters satisfy this constraint. These methods can be applied when a latent distribution parameters λ_i, associated with z_i controls the expectation of the distribution, such as the mean Gaussian parameters. In the following, we will assume that z_i are Gaussian-based, and denote μ_{z_i} their mean. Similar methods can be designed with other parameters with other distributions.

The rectification of the mean μ_{z_i} of Gaussian-based latent distributions (see (16)) is similar to the rectification of latent samples.

$$\underline{\mu}_{zi+1} = \max\left(\mu_{z_{i+1}}, \mu_{z_i}\right), \quad \forall i \in [\![1, n-1]\!] \tag{16}$$

This hard constraint guarantees that the expectation of the resulting distributions are ordered. However, rectifying latent distribution parameters can again lead to sub-optimal training. The encoder may not learn to output $\mu_{z_{i+1}} > \mu_{z_i} \ \ \forall i$, and may always rely on the rectification step to produce distributions that have ordered expectations, leading to $\mu_{z_{i+1}}$ and μ_{z_i} being always equal.

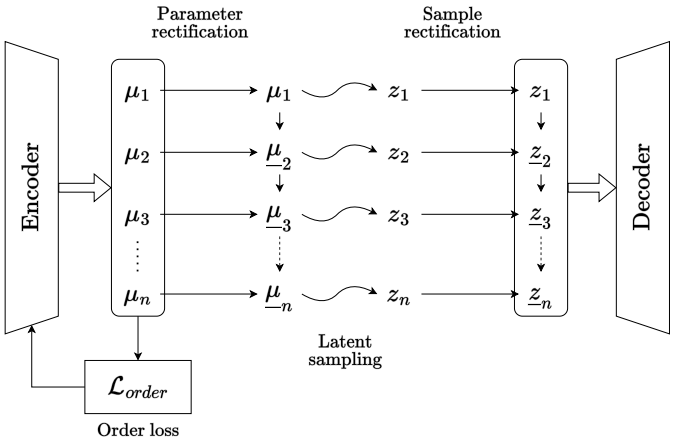


Fig. 6. Procedure of latent samples z_i ordering with maximum of latent distributions, with latent distribution parameters λ_i.

To ensure proper learning, we add a soft constraint in the form of a loss term in (17), that penalizes inference of disordered latent distribution parameters.

$$\mathcal{L}_{\text{order}} = \frac{1}{N}\sum_{i=1}^{N} \underline{\mu}_{z_i} - \mu_{z_i} \tag{17}$$

The *order loss* in (17) can be interpreted as an additional prior on latent distribution that the original KL term doesn't enforce.

Finally, using the maximum of consecutive variables to order the them does change their distribution (see appendix D for the density of the maximum of random variables). The prior distribution and the KL loss term can both be expected to become harder to derive for such latent distributions. In such case, we advocate for using the latent distribution without taking the ordering procedure into account in the computation of the prior and the KL term.

IV. Application: Inferring phenological parameters from NDVI time series

The interest of the proposed model inversion methodology is illustrated by a well-known parameter retrieval application. Specifically, the goal is to infer the probability distributions of the intrinsic phenological parameters from NDVI time series by considering a vegetation phenological model. This section presents the application of the proposed informed deep learning method in this specific application, which is denoted by pheno-VAE. The two data-sets used for training and validation purposes are also described here.

A. The phenological model as physics-based decoder

The NDVI quantifies land surface greenness and photosynthetic vegetation vigor [33]. It is derived from Near Infra-Red (NIR) and Red reflectances (R) of a land surface, and its expression is:

$$\text{NDVI} = \frac{\rho_{NIR} - \rho_R}{\rho_{NIR} + \rho_R} \in [-1, 1]\,. \tag{18}$$

This index is typically close to 1 for high densities of vegetation, close to 0 for bare soil, and negative for water.

TABLE I
Parameters of the double-logistic phenological model.

Variable	Description	Range $[a, b]$
M	Maximum of double logistic	[-1, 1]
m	Minimum of double logistic	[-1, 1]
sos	DOY[4] of *Start Of Season*, the start of NDVI growth	[-45, 410]
mat	DOY of *Maturity*, the end of NDVI growth	[-45, 410]
sen	DOY of *Senescence*, the start of NDVI decay	[-45, 410]
eos	DOY of *End Of Season*, end of NDVI decay	[-45, 410]

The characterization of the evolution of vegetation phenology using NDVI derived from remote sensing is widely addressed in the literature [34]–[36]. In general, the annual evolution of NDVI of some vegetation and crops can be well fitted with a double-logistic model [6], [37], [38]. The inversion of this *phenological model* from NDVI time series allows extracting *phenological parameters*, that typically characterize *phenophases* of the observed vegetation. The model we use here to characterize seasonal vegetation cycles on yearly time series is a 6-variable phenological model. This phenological model is described by the following equations:

$$\Omega_{\mathbf{z}}(t) = (M - m)\left(S_{sos,mat}(t) - S_{sen,eos}(t)\right) + m; \tag{19}$$

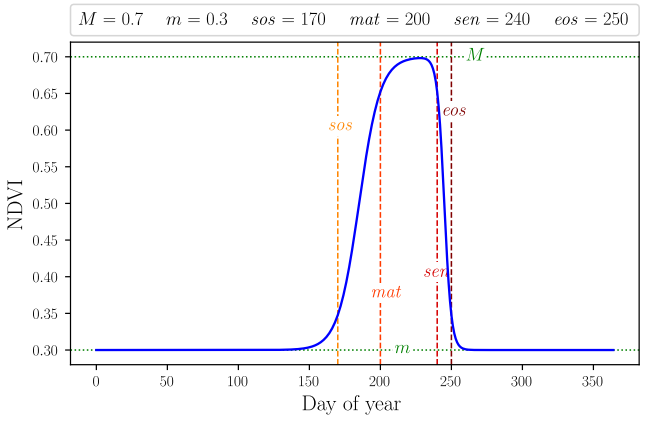
$$S_{\text{sos},mat}(t) = \left(1 + \exp\left(2\frac{sos + mat - 2t}{mat - sos}\right)\right)^{-1}; \tag{20a}$$

$$S_{sen,eos}(t) = \left(1 + \exp\left(2\frac{sen + eos - 2t}{eos - sen}\right)\right)^{-1}. \tag{20b}$$

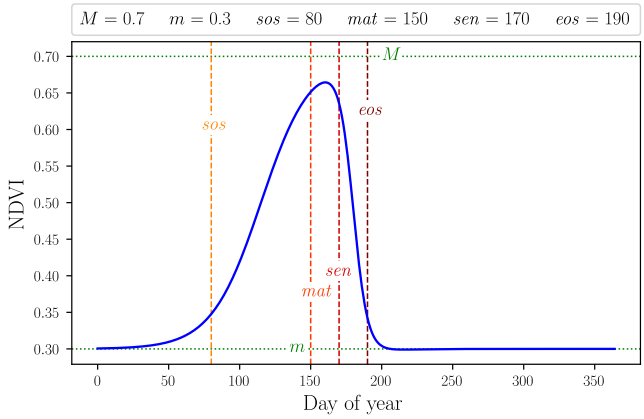
This model accounts for vegetation annual cycle, with a growth phase, a stagnation phase and a decay/harvest phase. The 6 *phenological parameters* $\mathbf{z} = (M, m, sos, mat, sen, eos)$ are described in Table I, and their effects on the model are shown in Fig. 7. Phenological parameters are all bounded. As observed, M and m have the same bounds as NDVI itself.

The range of *phenological dates* (*sos*, *mat*, *sen*, *eos*) are the days of a given calendar year, extended by 90 days. The 45 days considered before the 1st January and after the 31st December allows less restrictive estimations, and takes into account vegetation whose cycle started or ended outside the calendar year. This range is a prior knowledge about the data, like the double-logistic model itself.

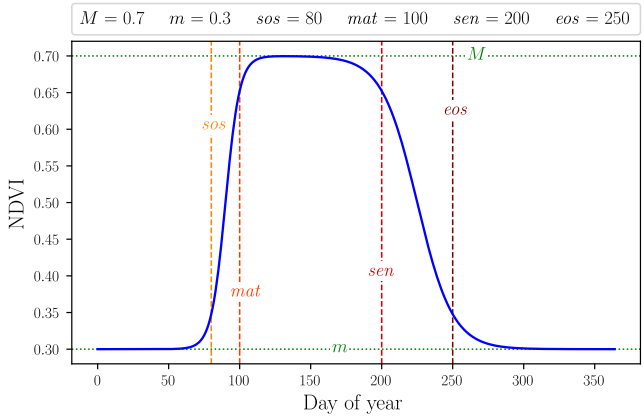
Following the architecture depicted in Fig. 3, the proposed pheno-VAE architecture proposes to use the double-logistic phenological model as an untrained, physical-based decoder. Each variable z_i of its 6-dimensional latent space is semantically bounded to a phenological parameter. The reconstruction term is computed as discussed in Section III-B. To take into account that phenological parameters are bounded, we propose Truncated Gaussians $\mathcal{TN}$ as latent distributions. The latent sampling process is performed with the inverse transform



(a)



(b)



(c)

Fig. 7. Examples of double logistic curves simulating different phenology for different vegetation covers, with different phenological parameters.

method described in (11). The phenological variables are ordered: $m < M$, and $sos < mat < sen < eos$, meaning the associated latent variables must be ordered in the same way. For that, the phenological dates are sampled as the maximum of all previous phenological dates, using the three strategies defined in Section III-D3: (i) the rectification of samples z_i in (15), (ii) the rectification of parameters μ_{z_i} of Truncated Gaussians in (16), (iii) the incorporation of an order term to the ELBO loss in (17).

As the latent variables of pheno-VAE are used as phenological model parameters, their distributions will be referred to as *phenological distributions*.

B. Data-sets

Two data-sets are used to evaluate the performances of pheno-VAE for phenological parameter retrieval. The first data-set is composed of real satellite observations of annual NDVI time series and is used for pheno-VAE training and qualitative validation. The second data set is composed of simulated crop NDVI profiles. The construction of this data-set is proposed for three main reasons: (i) to perform a quantitative evaluation of parameter retrieval on a large scale data-set, (ii) to assess the robustness of pheno-VAE to the noise of complex satellite observations, (iii) to compare the results of pheno-VAE against supervised methods. Examples of NDVI time series from both of data-sets are illustrated in Fig. 9.

1) S2 data-set: It is composed of 10^6 annual time series of pixels from 31TCJ Sentinel-2 tile[5] (Toulouse area in southern France) and it is available in [39]. The corresponding NDVI time series are computed from the spectral band 4 (Red) and 8 (Near Infra-Red). The resulting time series describe different land cover classes which can be associated to the class legend used in the CES OSO[6] land cover map [40]. Accordingly, a large number of time series do not represent vegetation classes following the double-logistic phenological model. Despite the availability of land cover class information, it must be remarked that such information is only used for validation purposes. Land cover classes do not intervene in the training procedure of pheno-VAE, and all samples are taken into account within a single training. The distribution of the land cover classes in the data-set is detailed in Table VIII of appendix A. The time series are acquired on irregular time intervals for two main reasons. Firstly, the two Sentinel-2 satellites have intersecting ground footprints and some locations get increased coverage. Secondly, cloud cover leads to inconsistent temporal sampling for each pixel on the ground (see Fig. 12 in appendix A). For each time series, a validity mask is available to denote the valid satellite observations. As pheno-VAE encoder learns from regular sampled time series, this mask is used to linearly interpolate raw time series to a common regular temporal grid.

2) Simulated data-set: The corresponding data set is composed by a large number of simulations obtained by the double-logistics model. A high number of combinations of input parameter values are generated by considering the input parameter ranges of Table II. As input parameters from simulations are known, this data-set allows us to compute quantitative metrics to validate the performances of pheno-VAE.

[5]Sentinel-2 images are projected onto 110 km × 110 km overlapping tiles aligned with NATO's Military Grid Reference System. The tile naming conventions indicates its position across the Earth. For more information, please see https://labo.obs-mip.fr/multitemp/the-sentinel-2-tiles-how-they-work/.

[6]From the french *Centre d'Expertise Scientifique sur l'Occupation des Sols*, meaning scientific expertise centre for land cover. It brings inter-laboratory teams under the Theia Data and Services centre for continental surfaces. https://www.theia-land.fr/en/homepage-en/

To generate synthetic time series, phenological parameters are firstly sampled from uniform distributions. The double-logistic model is then used to produce the corresponding NDVI temporal profiles. To simulate the irregular temporal sampling, binary validity masks of real S2 time series are considered. These masks are applied on simulated time series to select time series values at certain dates. To generate more realistic time series simulations, a Gaussian noise of randomly sampled standard deviation $\sigma_n \sim \mathcal{U}(0, 0.1)$ is added to the NDVI profile. It accounts for epistemic uncertainty, as no real time series is perfectly described by the phenological model. The resulting time series are finally interpolated at a regular 5-days time grid. The data generation procedure is depicted in Fig. 8.

The configuration of the parameter sampling procedure is detailed in the following. For σ, which represents the standard deviation of the noise in the observations, we will chose a maximum value of 0.1 which corresponds to 10% of the maximum expected range for NDVI values. For the minimum value of NDVI (m), we define the range between 0 (bare soil) and 0.4 (presence of vegetation). The maximum value of NDVI (M) is defined relative to the minimum value. In general, it can be considered that M is at least 0.3 higher than m for crop classes, and M can not be higher than 1.

A strategy is proposed to enforce the temporal order of the 4 dates characterizing the phenological stages. The idea is to consider that a the temporal parameter can be defined by its previous one. End of season (*eos*) is allowed to be right after Senescence and up to 90 days later. Senescence (*sen*) is defined in the same way with respect to maturity (*mat*) and *mat* follows the same rationale with respect to start of season (*sos*). For *sos* we would need to give a very wide prior in order to take into account winter and summer crops. Instead of doing that, we introduce an additional variable sos_i which will model the probability of summer crop. This probability is used to adjust the starting point of the interval of prior values for *sos*. We assume that the earliest *sos* for a winter crop is on day 30 (end of January) and that the earliest summer crop can have an *sos* of 120 (late April). sos_i and σ_n are additional variables of the generative process of synthetic data that will not be inferred during experiments.

Sampling of parameters for synthetic time series generation is summarized in Table II.

TABLE II
DISTRIBUTIONS OF REFERENCE PHENOLOGICAL PARAMETERS SAMPLED FOR NDVI TIME SERIES SIMULATION WITH THE DOUBLE-LOGISTIC MODEL.

Parameter	Sampling interval	Parameter	Sampling interval
m	$\mathcal{U}(0, 0.4)$	M	$\mathcal{U}(m, 1)$
sos_i	$\mathcal{U}(30, 120)$	*sos*	$\mathcal{U}(sos_i, sos_i + 90)$
mat	$\mathcal{U}(sos, sos + 90)$	*sen*	$\mathcal{U}(mat, mat + 90)$
eos	$\mathcal{U}(sen, \text{sen} + 90)$	σ_n	$\mathcal{U}(0, 0.1)$

Fig. 9 shows some simulated NDVI time series obtained by the proposed generation process. In it, the synthetic NDVI profiles are compared with real S2 time series describing three crop classes. As it can be observed, realistic simulated time series composed the synthetic data set. Even though the synthetic data-set is generated to be as realistic as possible, it is still different from the S2 data-set. Because of the uniform sampling of phenological dates in the synthetic data-set, there is more diversity in the phenology than the S2 data-set. On the one hand, the S2 data-set is biased by the samples that have been chosen among available real NDVI time series. All samples belong to the same S2 tile so NDVI time series of pixels of the same type are highly correlated, and cloud coverage similarly affects all time series. On the other hand, the synthetic data-set contains samples whose phenology that may not be frequent in reality, or even phenology types that don't exist. These differences will have to be taken into account in the interpretation of the results.

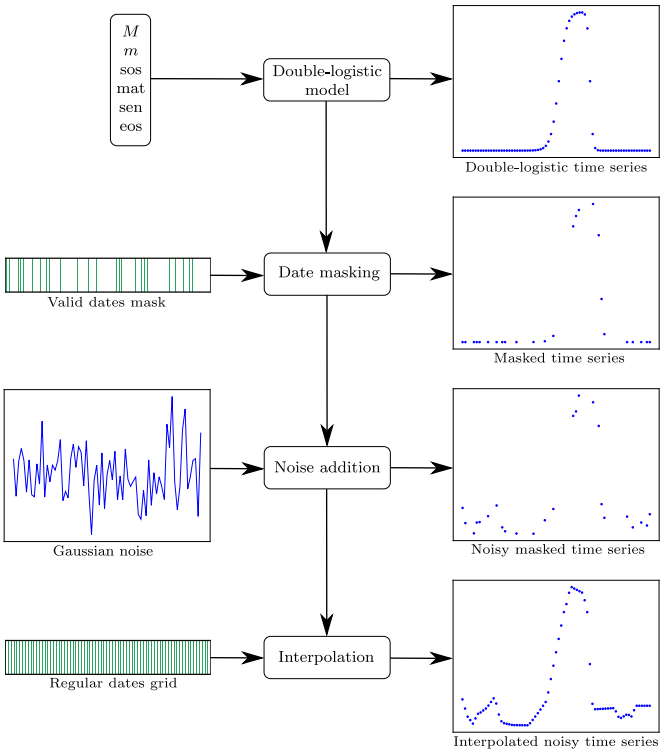


Fig. 8. Procedure of generation of a data-set of synthetic NDVI Time series.

C. Encoder of pheno-VAE

Pheno-VAE uses the encoder described in Fig. 10 corresponding to a multi-layer perceptron requiring regular (temporal sampled) NDVI time series of single pixels. This encoder architecture was chosen simple to show that satisfactory results can be obtained without using a complex encoder. Pheno-VAE's encoder outputs the parameters of truncated Gaussians μ_{z_i} and σ_{z_i} for each variable z_i. The support $[a_i, b_i]$ of each truncated Gaussian is set to $[0, 1]$. Each sample drawn from these distributions is scaled accordingly to the range described in Table I, using the procedure described in (13), before being transferred to the physics-based decoder.

The neural network is implemented using PyTorch. As there is no temporal encoding of time series, its input layer of size 73 is presented with annual NDVI time series sampled in a 5-days regular grid.

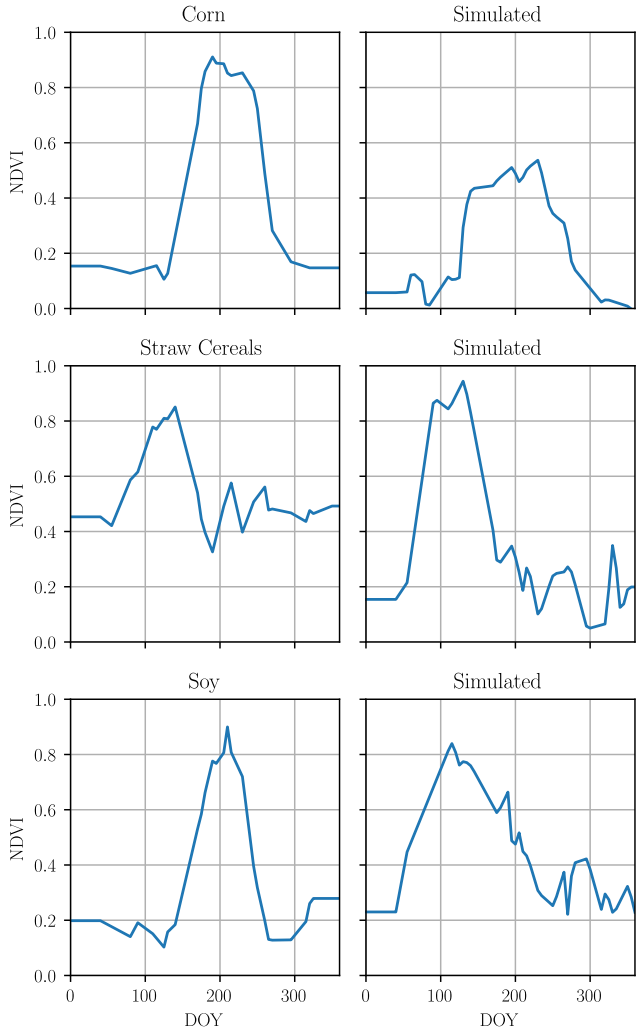


Fig. 9. NDVI time series of samples of S2 data-set (left) and simulated data-set (right)

D. Latent prior distribution and KL term in pheno-VAE

Because of the variety of training samples in both data-sets, in terms of phenology or even in terms of aleatoric and epistemic uncertainty, it is difficult to design a very restrictive prior. We chose a uniform distribution for all latent variables over their respective density support. The expression and derivation of the KL divergence between Truncated Gaussian and uniform distribution is provided in appendix E.

In practice, this loss promotes the inference of Truncated Gaussian posteriors with larger variances, while not penalizing their locations. Samples of the simulated and S2 data-sets have a wide variety of potential phenological parameters, and this loss doesn't promote any particular value for inference. In the S2 data-set, many samples don't have a phenology (buildings, mineral surfaces). For these time series, the reconstruction error should be high and variance of phenological parameters should increase to express epistemic uncertainty.

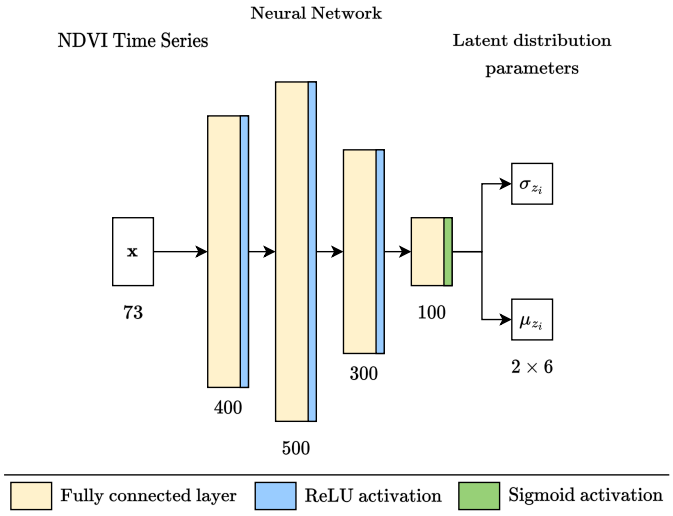


Fig. 10. Encoder architecture used in pheno-VAE, with 4 fully connected hidden layers with ReLU activation.

E. Loss of pheno-VAE

The loss functions minimized during the pheno-VAE training is composed by the next three terms :

$$\mathcal{L}_{pheno-VAE} = \mathcal{L}_{rec} + \beta\mathcal{L}_{kl} + \mathcal{L}_{order}. \tag{21}$$

The loss components are:

- $\mathcal{L}_{rec}$ the Gaussian NLL reconstruction loss,
- $\mathcal{L}_{kl}$ the KL divergence between the Truncated Gaussian latent variables and the uniform prior.
- $\mathcal{L}_{order}$ term to promote ordered latent variables.

In practice, $\mathcal{L}_{order}$ converges to zero very fast, leaving only the two other terms in most of the training process. There is a tension between the two remaining terms: the reconstruction loss improves the quality of the reconstructed time series, and the Kullback-Leibler divergence acts as a regularizer of the latent space. The balance between these two terms is adjusted by coefficient β for the KL term as proposed in β-VAE [41]. The influence of the hyper-parameter β is studied in the following section.

V. Experiments

In this section, the experimental setup and the evaluation metrics used to evaluate the quality of the inferred phenological parameters are described, then the obtained results are presented.

A. Computing Environment

All computations performed for this work were executed using the CNES's High Performance Computing Center infrastructure. Specifically, we used the following hardware to run our experiments :

- CPU model: Intel(R) Xeon(R) CPU E5-2698 v4
- GPU model: NVIDIA Tesla V100-SXM2-32GB
- RAM: 64 GB.

B. Experimental setup

Different experiments are carried out to assess the performances of pheno-VAE. Firstly, the reconstructions of NDVI time series obtained by pheno-VAE trained on S2 data-set are visually evaluated. Secondly, a quantitative assessment of pheno-VAE is performed by inferring the phenological parameters of the simulated data-set. The statistical assessment is carried out through two experiments : (i) a first experiment to evaluate the influence of β and (ii) a second evaluation to compare pheno-VAE with different standard parameter retrieval algorithms. For the comparison, a Neural Network Regression (NNR) method, a Curve Fitting (CF) algorithm and a Markov chain Monte Carlo (MCMC) algorithm and allowing the estimation of parameter uncertainties are considered. All these methods perform the inversion of the phenological model on the NDVI time series of single pixels.

The NNR method is proposed as a hybrid physics-assisted machine-learning solution for the output regression problem. The supervised training of this network is performed using the simulated data-set in order to estimate the mean and variance of the Truncated Gaussian distributions associated to the 6 phenological parameters. The training of the supervised regression algorithm is performed by applying the Negative Log-Likelihood of Ordered Truncated Gaussians (see appendix D). Knowing the phenological parameter values of the synthetic data sets, the NLL compares the phenological distributions estimated from the regression algorithm against the known phenological parameters. To ensure that model complexity doesn't influence comparative results, the architecture of the regression network is identical to that of pheno-VAE (see Fig. 10).

Concerning the MCMC, this algorithm is typically used to sample from a probability distribution. The main advantage of this method is that it allows to draw samples where the next sample is dependent on the existing sample, called a Markov Chain. MCMC can be used in Bayesian inference by using it to sample the intractable posterior distribution, i.e. the parameters of the model to invert. Following the methodology of [15], we use Hamiltonian Monte Carlo as per the NUTS algorithm [42] as implemented in the NumPyro library [43], [44].

To implement Bayesian inference through MCMC we need to define the likelihood for the observed data. To measure the goodness of the model fit to the data, the double-logistic function is used. At inference, NDVI time series irregularly sampled are injected into MCMC algorithm. As prior distributions, we choose the same uniform distributions described in Table II.

The CF algorithm solves a non linear least squares problem for each NDVI time series, with a trust region reflective algorithm [45]. This method can take the boundaries of the parameters into account. Although it is not a Bayesian approach, the CF algorithm outputs along with the predicted parameters, a covariance matrix that can be used to estimate prediction intervals. Unfortunately, as the inversion is frequently ill-conditioned, the estimated parameters covariances often diverge. Thus we discard confidence intervals estimation with this method. Contrary to other inversion methods presented here, the CF requires an initial guess z_0 on the parameters (i.e. it requires more prior information). The initial guess we used to fit the phenological model on NDVI time series is detailed in Table III. This method was implemented by using the `curve_fit` function of python's scipy.optimize library.

TABLE III
INITIAL GUESS OF THE PHENOLOGICAL PARAMETERS FOR THE CURVE FITTING ALGORITHM.

Parameter	Initial guess	Parameter	Initial guess
m	$\max(y_i)$	M	$\max(y_i)$
sos	$\arg\max(y_i) - 30$	*mat*	$\arg\max(y_i) - 15$
sen	$\arg\max(y_i) + 15$	*eos*	$\arg\max(y_i) + 30$

It should be noted that CF, MCMC and NNR performances are provided as an upper bound for parameter retrieval performances. The limitations of both strategies for large-scale parameter retrieval applications have been previously described in the introduction section. Besides NNR and MCC methodologies, two training pheno-VAE scenarios are considered to evaluate the influence of the training data-set. The two trained pheno-VAE configurations are trained on the S2 data-set, and the other with the synthetic data-set. The characteristics of the different methods used for the qualitative assessment experiment are summarized in Table IV.

C. Evaluation metrics

The three metrics described in the following are used to evaluate the accuracy of the retrieved parameters and their corresponding uncertainties. The synthetic validation data-set is composed of 10000 samples.

1) Point estimate inference error: The Mean Absolute Error (MAE) between the known parameter value and the obtained prediction is computed. As proposed methods predict probabilistic outputs, the prediction value obtaining the best MAE is considered. The distribution mode is proposed for the NNR, pheno-VAE (S2 & sim), and the median for MCMC. As MAE is sensitive to outliers, box-plots of the absolute errors are provided in appendix C-B.

2) Prediction interval metrics: Prediction intervals are at the core of uncertainty quantification [46], [47]. In this work, we estimate the prediction intervals of phenological variables from the inferred distributions (i.e. the distribution approximate for MCMC, the truncated Gaussian distribution inferred with NNR and in the latent space of pheno-VAE). To assess the quality of these intervals, two prediction intervals metrics widely used in the literature are considered [48], [49]:

- The Mean Prediction Interval Width (MPIW). Because it is sensitive to outliers, box-plots of the prediction interval widths are provided in appendix C-E.
- The Prediction Interval Coverage Probability (PICP). It measures the frequency of the model parameters true value being inside the prediction interval, and its value should be as close to the confidence level as possible.

In the following, these metrics are computed for prediction intervals with a selected 90% confidence level, by using the 5^{th}-95^{th} percentile intervals. Results obtained with different

TABLE IV
CHARACTERISTICS AND HYPER-PARAMETERS OF EACH EXPERIMENTS INFERENCE METHODS.

Method	Supervised	Training	Optimizer	Batch size	Learning Rate	Epochs	Latent samples	Point estimate	Parameter distribution
CF	✗	✗	✗	✗	✗	✗	✗	Deterministic	✗
MCMC	✗	✗	✗	✗	✗	✗	✗	Median	Full posterior approximate
NNR	✓	Simulated Data-set	Adam	2048	5.10^{-4}	500	✗	Mode	Truncated Gaussian
pheno-VAE (Sim)	✗	Simulated Data-set	Adam	2048	5.10^{-4}	200	10	Mode	Truncated Gaussian
pheno-VAE (S2)	✗	S2 Data-set	Adam	2048	5.10^{-4}	200	10	Mode	Truncated Gaussian

confidence levels are shown in appendix C-D and C-C. The equations of these metrics are provided in appendix.

The three evaluation metrics are computed by using a K-fold cross-validation procedure, in which the data-set is divided into K folds. In each round, a model is trained using K − 1 of the folds as training data and tested on the remaining set. Metrics are then measured by averaging the performance values computed on each subset (K models). This strategy is applied to validate deep learning approaches. For MCMC, metrics are independently obtained on K subsets of the total data-set. The averages and standard deviations of the results on those subsets are computed. In the following, K is equal to 6.

D. Evaluation of the reconstruction results

To assess the performances of pheno-VAE, a visual evaluation is presented in Fig. 11. This figure shows the reconstruction of different S2 NDVI time series obtained by the pheno-VAE model trained on S2 data. For each example, the estimated phenological parameter distributions are also illustrated. In most cases shown here, the setting $\beta = 0$ imposes that no prior information from the data-set is incorporated. This is different from our uniform prior that assumes that phenological variables are evenly distributed over their possible range.

In general, the error and variance of reconstructions are both low for temporal profiles well-characterized by the phenological model. The estimated phenological distributions seem well centered on likely phenological parameters. Fig. 11a shows NDVI time series of a pixel of corn, the inferred phenological distributions and the reconstruction of its mode. The reconstruction curve is observed to accurately match the original time series. The distributions of phenological dates characterize well the growth and decay phases of this summer crop.

The influence of β can be evaluated by comparing the results observed on Fig. 11(a) and 11(b). The same NDVI time series of a corn pixel is taken as input by two pheno-VAE models with different values of β. The modal reconstructions are very similar. With increasing β, the phenological distributions widen, and the variance of reconstructions increases. This is coherent with the influence of the KL loss terms, that discourages narrow latent densities. With both results well matching the original NDVI time series, the choice of β is to be made considering the prediction interval metrics.

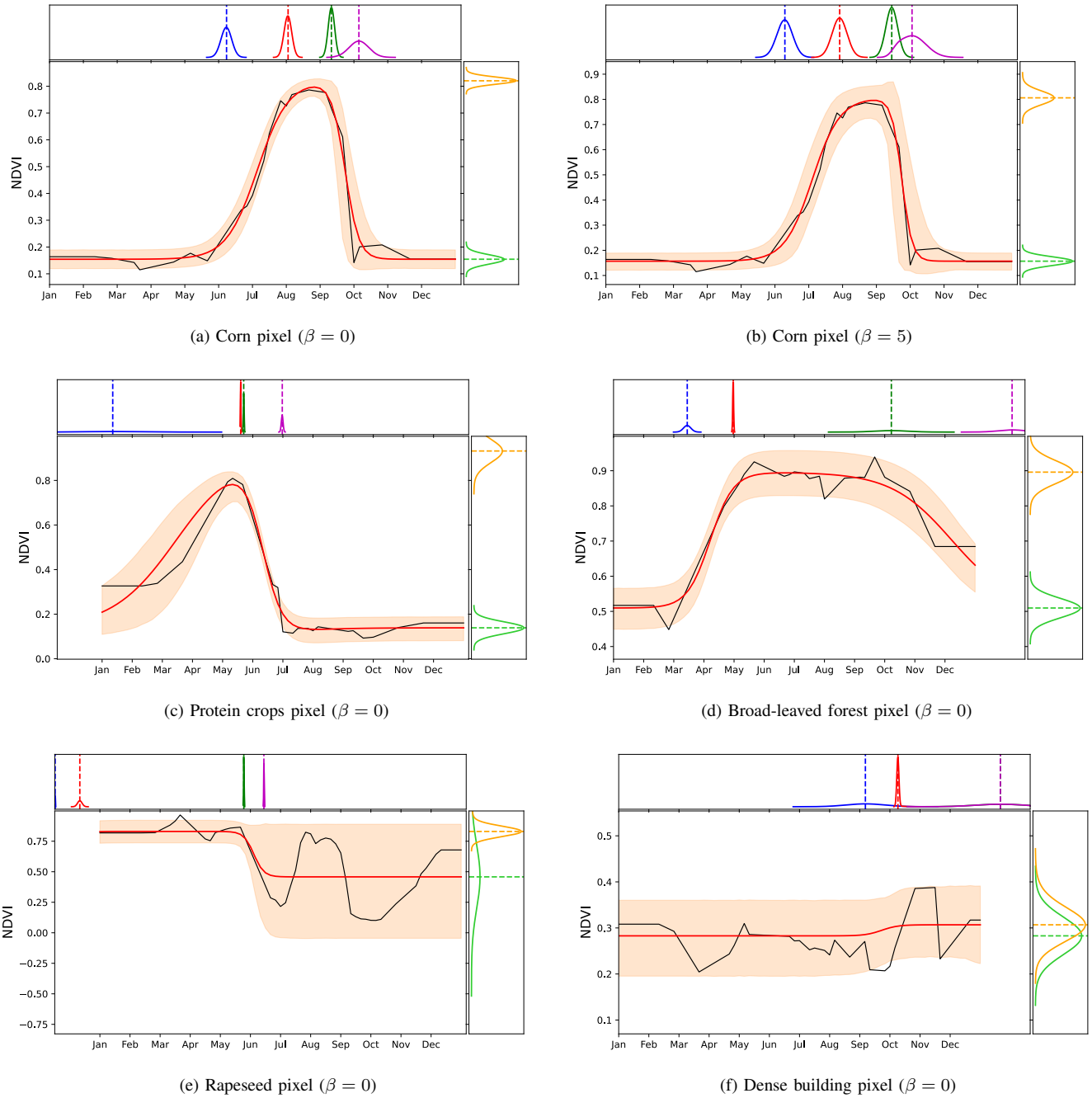
On Fig. 11(c), a protein crop time series shows how the presence of data gaps can lead to bad phenological parameter estimation. In this figure, the phenological cycle is easily identifiable. However, bad weather in winter led to a lack of data points for the first two months, and the backward extrapolation of points at pre-processing has kept the NDVI artificially constant, at a higher value than after harvest. As the encoder of pheno-VAE doesn't take into account the temporal information, here reconstruction is disrupted by the gap-filling step. This extrapolation artifact made the input time series not well described by the phenological model at the beginning of the year. The start of season estimate is inaccurate, yet the distribution large spread indicates greater uncertainty. This bad inference of the start of season seems to have prevented a good estimation of the maturity date as well, with this time a narrow distribution. Nonetheless the senescence and end of season seem well inferred. Similarly with a broad-leaved forest time series (Fig. 11(d)), senescence and end of season distributions are not well positioned due to interpolated data points at the end of year. These results show that the gap-filling pre-processing task can lead to wrong parameter estimations when long data gaps include key phenological dates. This highlights the need for encoders that don't rely on interpolated inputs. However, that would be out of the scope of our current contribution.

In Fig. 11(e), there are several crops in the pixel, and the NDVI time series shows several phenological cycles. As the model can only take one cycle into account, it only fits the largest, and takes the average of the remaining signal. The distribution of the minimum of NDVI is very large, indicating uncertainty.

In Fig. 11(f), the phenological model doesn't suit at all the NDVI time series of a dense building pixel. Therefore, reconstruction errors are high. However phenological distribution variances increase to take this epistemic uncertainty into account. These results show that large uncertainties could be associated to the model discrepancy with the data.

Another remark is that, inferred marginal phenological distributions sometimes show significant overlap. This highlights the interest of the proposed order constraints on the latent distributions, as reconstructions are consistent with the phenological model, and variables constraints are always respected.

More reconstruction examples are available in appendix B.



(a) Corn pixel ($\beta = 0$)

(b) Corn pixel ($\beta = 5$)

(c) Protein crops pixel ($\beta = 0$)

(d) Broad-leaved forest pixel ($\beta = 0$)

(e) Rapeseed pixel ($\beta = 0$)

(f) Dense building pixel ($\beta = 0$)

Fig. 11. Reconstruction and phenological parameters distributions from the encoding of the NDVI time series by pheno-VAE trained on S2 data-set. Central quadrants, S2 NDVI time series (black), reconstructions from the modes of phenological parameters distributions (red), and reconstruction 5^{th}-95^{th} prediction interval - Upper quadrants: Truncated Gaussian distributions of the 4 phenological dates, *sos* (blue), *mat* (red), *sen* (dark green), *eos* (magenta) - Right quadrants: Truncated Gaussian distributions of M (orange), and m (light green) - Upper and right quadrants: distribution densities are in solid lines, distribution modes are in dashed lines.

E. Influence of the KL loss term on pheno-VAE performances

The impact of the KL term is studied by comparing results obtained by using different β values. In this experiment, pheno-VAE model is trained with samples from the S2 data-set. The prediction interval metrics presented here are derived for a confidence level of $1 - \alpha = 0.9$.

As previously observed, the KL term tends to increase the dispersion of the phenological parameters distributions. The MPIW (Table V(c)) and PIW (Fig. 20)) increases for the phenological dates along with β and consequently the PICP (Table V(b)) also increases.

The MAE results (Table V(a)) tend to increase along with β, decreasing performance, although the distributions of the absolute errors (Fig. 14) only worsen significantly above a certain threshold of β. These results corroborate that the hyper-parameter β must be selected by using an independent validation data-set. For the prediction intervals to be informative, the KL term needs to be high enough, while keeping it below a certain threshold ensures that precision is acceptable.

TABLE V
EVALUATION PERFORMANCES OBTAINED ON A SIMULATED DATA-SET FOR DIFFERENT PHENO-VAE MODELS TRAINED ON THE S2 DATA-SET, AND FOR VARIOUS KL LOSS COEFFICIENTS β. PREDICTION INTERVALS ARE DERIVED FROM PHENOLOGICAL DISTRIBUTIONS WITH A CONFIDENCE LEVEL $1-\alpha=0.9$.

Exp.	M	m	*sos*	*mat*	*sen*	*eos*
pheno-VAE (S2, $\beta=0$)	$\mathbf{0.05 \pm 0.00}$	$\mathbf{0.02 \pm 0.00}$	$\mathbf{11.13 \pm 0.46}$	$\mathbf{10.22 \pm 0.08}$	$\mathbf{11.01 \pm 0.47}$	$\mathbf{13.35 \pm 0.52}$
pheno-VAE (S2, $\beta=1$)	$\mathbf{0.05 \pm 0.00}$	$\mathbf{0.02 \pm 0.00}$	11.82 ± 0.27	10.38 ± 0.33	11.61 ± 0.65	13.48 ± 0.69
pheno-VAE (S2, $\beta=2$)	$\mathbf{0.05 \pm 0.00}$	$\mathbf{0.02 \pm 0.00}$	11.93 ± 0.60	10.58 ± 0.25	12.15 ± 0.60	14.75 ± 0.97
pheno-VAE (S2, $\beta=5$)	0.07 ± 0.00	$\mathbf{0.02 \pm 0.00}$	14.87 ± 0.21	14.37 ± 0.61	18.37 ± 0.75	18.69 ± 0.47

(a) Mean Absolute Error

Exp.	M	m	*sos*	*mat*	*sen*	*eos*
pheno-VAE (S2, $\beta=0$)	$\mathbf{0.67 \pm 0.01}$	0.95 ± 0.01	0.34 ± 0.05	0.25 ± 0.03	0.34 ± 0.04	0.58 ± 0.02
pheno-VAE (S2, $\beta=1$)	0.60 ± 0.01	0.95 ± 0.01	0.53 ± 0.02	0.48 ± 0.02	0.55 ± 0.01	0.71 ± 0.02
pheno-VAE (S2, $\beta=2$)	0.61 ± 0.02	0.94 ± 0.01	0.64 ± 0.02	0.56 ± 0.01	0.64 ± 0.01	0.76 ± 0.03
pheno-VAE (S2, $\beta=5$)	0.63 ± 0.03	$\mathbf{0.92 \pm 0.01}$	$\mathbf{0.77 \pm 0.01}$	$\mathbf{0.69 \pm 0.02}$	$\mathbf{0.69 \pm 0.02}$	$\mathbf{0.83 \pm 0.01}$

(b) Prediction Interval Coverage Probability

Exp.	M	m	*sos*	*mat*	*sen*	*eos*
pheno-VAE (S2, $\beta=0$)	0.12 ± 0.01	0.13 ± 0.00	$\mathbf{14.69 \pm 2.85}$	$\mathbf{8.81 \pm 1.11}$	$\mathbf{13.75 \pm 1.01}$	$\mathbf{30.60 \pm 1.83}$
pheno-VAE (S2, $\beta=1$)	$\mathbf{0.11 \pm 0.00}$	$\mathbf{0.12 \pm 0.00}$	22.97 ± 1.38	18.24 ± 1.05	23.35 ± 1.18	36.60 ± 2.38
pheno-VAE (S2, $\beta=2$)	$\mathbf{0.11 \pm 0.00}$	$\mathbf{0.12 \pm 0.00}$	27.93 ± 1.54	22.81 ± 0.75	28.43 ± 1.53	43.30 ± 3.10
pheno-VAE (S2, $\beta=5$)	0.16 ± 0.00	$\mathbf{0.12 \pm 0.00}$	41.79 ± 1.64	38.24 ± 1.65	42.18 ± 1.36	59.64 ± 2.30

(c) Mean Prediction Interval Width

Also, different performances are obtained for the different phenological parameters. The minimum of NDVI m is the best estimated parameter, as with simulated time series, a large part of available data points are around the value of the minimum — although, it is so well estimated that its prediction interval almost always contains it, overshooting the PICP $= 1-\alpha$ target. The parameter M is more challenging to estimate than m. The value of the true maximum of the phenological model can differ from the parameter M when *mat* and *sen* are close. The highest errors are obtained on phenological dates, most certainly because of the gap-filling problem highlighted with reconstruction results (such as with Fig. 11(c) and 11(d)). This limitation is more visible in MPIW values obtained for *sos* and *eos* than *mat* and *sen*. This is because the pheno-VAE is confronted with more severe extrapolation aberrations at both ends of the time series than in the middle, where interpolation is better, with higher temporal availability in the original time series.

In the following, the setting $\beta = 2$ will be used, as it increases the PICP without degrading too much the MPIW and the MAE.

F. Quantitative assessment of pheno-VAE

Quantitative results obtained by pheno-VAE trained on S2 data-set, pheno-VAE trained on the synthetic data-set, MCMC and NNR by inferencing the phenological distributions of the simulated data-set are compared here. Obtained results are presented in Table VI.

Best overall performances are obtained by the MCMC, for which the distribution of absolute errors is the lowest (Fig. 14), despite having a little higher MAE (Table VI(a)) than NNR. MCMC also attains PICP that is close to the confidence level α (Table VI(b) and Fig. 17), with prediction intervals significantly narrower than other presented methods. Phenological distribution inference is not limited by a distribution family prior and directly samples phenological distributions, contrary to the other methods studied here. It is also not affected by missing data gaps because MCMC do not require regularly temporal input data. The results of MCMC could be improved by increasing the number of distribution samples and steps, at the expense of greater computation costs. Despite the promising MCMC results, its computing time required is much longer for MCMC than deep learning methods (see Table VII). It justifies why such approach can not be applied on operational parameter retrieval applications.

NNR, has absolute errors that are a little higher and larger prediction intervals, however it has the best PICP, that is the closest to the confidence level α for all phenological variables. Those good results are expected, considering that it is a supervised method, with the training data-set being very similar to the testing data-set. Furthermore its loss doesn't rely on reconstruction, and therefore isn't affected by the irregular temporal sampling of real S2 time series.

The CF approach predicts phenological parameters with a MAE between that of MCMC and NNR, except for *mat* and *sen* which are on par with the inference of pheno-VAE. However we observed that this method was less reliable than the other presented here, as it didn't converge to a solution for about 5% of the time series (the results presented in Table VI(a) excluded those failed predictions).

The results of pheno-VAE are less good than MCMC and NNR. It has higher MAE, and despite similar prediction interval sizes, it underestimates uncertainty with lower PICP. Results also show different behaviors for the two pheno-VAE trained on different data-sets. As expected, slightly better results are obtained when pheno-VAE is trained on simulated

TABLE VI
EVALUATION PERFORMANCES OBTAINED ON A SIMULATED DATA-SET FOR DIFFERENT EXPERIMENTS OF INVERSION OF THE PHENOLOGICAL MODEL. PREDICTION INTERVALS ARE DERIVED FROM PHENOLOGICAL DISTRIBUTIONS WITH A CONFIDENCE LEVEL $1-\alpha = 0.9$.

Exp.	M	m	*sos*	*mat*	*sen*	*eos*
pheno-VAE (S2, $\beta = 2$)	0.05 ± 0.00	0.02 ± 0.00	11.93 ± 0.60	10.58 ± 0.25	12.15 ± 0.60	14.75 ± 0.97
pheno-VAE (Sim, $\beta = 2$)	0.06 ± 0.00	0.02 ± 0.00	8.89 ± 0.53	10.51 ± 0.49	10.59 ± 0.52	9.23 ± 0.26
MCMC	$\mathbf{0.03 \pm 0.00}$	0.02 ± 0.00	7.18 ± 0.70	9.57 ± 0.95	9.93 ± 1.00	10.42 ± 1.18
NNR	0.04 ± 0.00	$\mathbf{0.01 \pm 0.00}$	$\mathbf{6.69 \pm 0.03}$	$\mathbf{7.54 \pm 0.05}$	$\mathbf{6.91 \pm 0.05}$	$\mathbf{6.70 \pm 0.07}$
CF	0.07 ± 0.00	$\mathbf{0.01 \pm 0.00}$	7.58 ± 1.07	11.74 ± 1.20	10.75 ± 1.20	7.37 ± 1.25

(a) Mean Absolute Error

Exp.	M	m	*sos*	*mat*	*sen*	*eos*
pheno-VAE (S2, $\beta = 2$)	0.61 ± 0.02	0.94 ± 0.01	0.64 ± 0.02	0.56 ± 0.01	0.64 ± 0.01	0.76 ± 0.03
pheno-VAE (Sim, $\beta = 2$)	0.67 ± 0.01	0.99 ± 0.00	0.67 ± 0.05	0.60 ± 0.01	0.66 ± 0.01	0.77 ± 0.02
MCMC	0.89 ± 0.01	0.86 ± 0.01	0.84 ± 0.01	0.85 ± 0.01	0.83 ± 0.01	0.83 ± 0.01
NNR	$\mathbf{0.90 \pm 0.01}$	$\mathbf{0.90 \pm 0.01}$	$\mathbf{0.89 \pm 0.00}$	$\mathbf{0.89 \pm 0.00}$	$\mathbf{0.89 \pm 0.01}$	$\mathbf{0.88 \pm 0.00}$

(b) Prediction Interval Coverage Probability

Exp.	M	m	*sos*	*mat*	*sen*	*eos*
pheno-VAE (S2, $\beta = 2$)	$\mathbf{0.11 \pm 0.00}$	$\mathbf{0.12 \pm 0.00}$	27.93 ± 1.54	$\mathbf{22.81 \pm 0.75}$	28.43 ± 1.53	43.30 ± 3.10
pheno-VAE (Sim, $\beta = 2$)	0.14 ± 0.01	0.14 ± 0.00	$\mathbf{21.02 \pm 0.76}$	23.25 ± 1.32	27.09 ± 1.16	25.23 ± 0.80
MCMC	0.13 ± 0.01	0.05 ± 0.00	22.13 ± 1.75	25.03 ± 1.94	$\mathbf{22.74 \pm 1.79}$	$\mathbf{21.50 \pm 2.29}$
NNR	0.16 ± 0.00	0.06 ± 0.00	27.70 ± 0.30	29.91 ± 0.25	27.81 ± 0.43	26.36 ± 0.40

(c) Mean Prediction Interval Width

data. A greater performance drop is observed for *eos*. This is because of a discrepancy between both data-sets. In the simulated data-set, there is more diversity in the phenological parameters, because of the uniform sampling to generate it. Even if real validity masks from the S2 data-set are used, they are not correlated to phenology, as it is the case for real data. In the S2 data-set, a smaller diversity of combinations of phenological variables is available. In this data-set, the end of season of real crops can happen when there are clouds, more than in the simulated data-set.

The drop in performances is much less significant compared to regression and MCMC, despite training on samples that don't follow the phenological model. The pheno-VAE trained on the synthetic data-set benefits from being evaluated on a similar simulated data-set. This unfair advantage could be mitigated by evaluating the performances of pheno-VAE on real Sentinel-2 NDVI time series data-set, with available ground truth of phenological stages. Unfortunately, such a data-set was not available to us at the time of this study.

MCMC and NNR show similar performances, despite being very different methods. This hints that given the simulated data-set and the double-logistic model, there is not much performance improvement to expect from the inference experiment, even with other setups. The regression yields on phenological dates 7-day MAE, with 90% PICP and 28 days MPIW. These are good results considering irregularly sampled time series that are interpolated to a 5-day grid. For pheno-VAE to get performances closer to this, there is a need to improve on the ability of the encoder neural network to take temporal structure of time series into account. To minimize the impact of the gap-filling pre-processing step, different solutions could be considered. For instance, the reconstruction loss could be modified to only take valid observations into account. The encoder network architecture could be replaced to allow to learn from irregularly sampled time series such as with transformers.

TABLE VII
APPROXIMATE TRAINING AND INFERENCE TIME FOR EACH SETUP ON COMPUTING ENVIRONMENT DESCRIBED IN SECTION V-A

Method	CF	MCMC	NNR	pheno-VAE (Sim)	pheno-VAE (S2)
GPU usage	✗	✗	✓	✓	✓
Training	✗	✗	15 min	15 min	15 min
Inference per time series	10^{-4} s	10 s	10^{-5} s	10^{-5} s	10^{-5} s

G. Ablation study of the latent distribution maximum sampling techniques

An ablation study for the strategy presented to incorporate temporal structure in latent variables is performed with pheno-VAE. The three proposed sub-tasks are evaluated : the μ-rectification in (16), latent samples rectification in (15), and the order loss in (17). When any of these steps is removed, we observe that training convergence takes longer time. It also often leads to sub-optimal models that only order distributions by making them identical. Moreover, simply removing the latent sample rectification leads the pheno-VAE to infer latent model parameters that fit the data but no longer have physical meaning (with for instance the *sos* date being after the *eos* date).

VI. Conclusion

In this paper, we have proposed a new physics-guided deep learning probabilistic methodology to invert physical models. Different strategies are presented to incorporate physical knowledge in VAE by considering physical-based decoders. Semantic latent variables bound to physical model parameters have been learned by incorporating prior knowledge and order constraints in the learning process. Monte Carlo sampling of the latent space was introduced to generate a reconstruction distribution from deterministic decoders. The classical pair of prior and posterior distributions was modified to better represent the physics of the problem. Order constraints were added to better model the properties of physical variables in a semantic latent space. A new KL loss term was proposed, whose weight in the loss enable to adjust the performance of the model. The training is robust to samples which do not correspond to the physical model (pixels without vegetation). The feasibility and the interest of the proposed methodology has been corroborated through a well-known remote sensing inverse problem, the phenological parameter retrieval from Sentinel-2 NDVI time series.

Despite using a simple neural network architecture, preliminary results are encouraging. Enhancing the encoder architecture with inductive biases taking into account the temporal structure of the data (attention mechanisms, recurrent architectures) could improve the inference error and predicted prediction intervals that fall behind other methods in the current configuration. Furthermore, the exploitation of the spatial context of satellite data may improve parameter retrieval on individual pixels. As designing models that can simulate complete landscapes is challenging, this could be performed by using dedicated deep learning architectures, such as convolutional neural networks. Applying the methodologies to different models of more complex data will be the focus of future research efforts. The presented informative deep learning strategy could be an important step toward the large scale production of vegetation status indicators.

In an attempt to enable reproducible research, our implementation of the methods developed in this paper are available at the following: https://gitlab.cesbio.omp.eu/zerahy/pheno-VAE.git.

Acknowledgments

The authors would like to thank CNES for the provision of its high performance computing (HPC) infrastructure to run the experiments presented in this paper and the associated help. We also thank the ANR-JCJC-20-CE23-0003 DeepChange project for funding this research. We are especially grateful to Mathieu Fauvel and Julien Michel for their feedback on a preliminary version of this paper.

References

[1] F. Sarvia, S. De Petris, and E. Borgogno-Mondino, "Mapping Ecological Focus Areas within the EU CAP Controls Framework by Copernicus Sentinel-2 Data," *Agronomy*, vol. 12, no. 2, 2022, ISSN: 2073-4395. DOI: 10.3390/agronomy12020406. [Online]. Available: https://www.mdpi.com/2073-4395/12/2/406.

[2] J. Verrelst, Z. Malenovský, C. van der Tol, G. Camps-Valls, J.-P. Gastellu-Etchegorry, P. Lewis, P. R. J. North, and J. F. Moreno, "Quantifying vegetation biophysical variables from imaging spectroscopy data: A review on retrieval methods," *Surveys in Geophysics*, vol. 40, pp. 589–629, 2018.

[3] J. Verrelst, J. P. Rivera, F. Veroustraete, J. Muñoz-Marí, J. G. Clevers, G. Camps-Valls, and J. Moreno, "Experimental sentinel-2 lai estimation using parametric, non-parametric and physical retrieval methods – a comparison," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 108, pp. 260–272, 2015, ISSN: 0924-2716. DOI: https://doi.org/10.1016/j.isprsjprs.2015.04.013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0924271615001239.

[4] G. Misra, F. Cawkwell, and A. Wingler, "Status of phenological research using sentinel-2 data: A review," *Remote Sensing*, vol. 12, no. 17, 2020, ISSN: 2072-4292. DOI: 10.3390/rs12172760. [Online]. Available: https://www.mdpi.com/2072-4292/12/17/2760.

[5] M. Meroni, R. d'Andrimont, A. Vrieling, D. Fasbender, G. Lemoine, F. Rembold, L. Seguini, and A. Verhegghen, "Comparing land surface phenology of major european crops as derived from sar and multispectral data of sentinel-1 and-2," *Remote sensing of environment*, vol. 253, p. 112 232, 2021.

[6] L. Zeng, B. D. Wardlow, D. Xiang, S. Hu, and D. Li, "A review of vegetation phenological metrics extraction using time-series, multispectral satellite data," *Remote Sensing of Environment*, vol. 237, p. 111 511, Feb. 2020, ISSN: 0034-4257. DOI: 10.1016/j.rse.2019.111511. [Online]. Available: https://doi.org/10.1016/j.rse.2019.111511.

[7] T. N. Matongera, O. Mutanga, M. Sibanda, and J. Odindi, "Estimating and monitoring land surface phenology in rangelands: A review of progress and challenges," *Remote Sensing*, vol. 13, no. 11, 2021, ISSN: 2072-4292. DOI: 10.3390/rs13112060. [Online]. Available: https://www.mdpi.com/2072-4292/13/11/2060.

[8] H. Xu, W. He, L. Zhang, and H. Zhang, "Unsupervised spectral–spatial semantic feature learning for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–14, 2022. DOI: 10.1109/TGRS.2022.3159789.

[9] J. Verrelst, J. P. Rivera, G. Leonenko, L. Alonso, and J. Moreno, "Optimizing lut-based rtm inversion for semiautomatic mapping of crop biophysical parameters from sentinel-2 and -3 data: Role of cost functions," *IEEE Transactions on Geoscience and Remote Sensing*,

vol. 52, no. 1, pp. 257–269, 2014. DOI: 10.1109/TGRS.2013.2238242.

[10] D. H. Svendsen, L. Martino, M. Campos-Taberner, F. J. García-Haro, and G. Camps-Valls, "Joint gaussian processes for biophysical parameter retrieval," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 3, pp. 1718–1727, 2018. DOI: 10.1109/TGRS.2017.2767205.

[11] J. Estévez, M. Salinero-Delgado, K. Berger, L. Pipia, J. P. Rivera-Caicedo, M. Wocher, P. Reyes-Muñoz, G. Tagliabue, M. Boschetti, and J. Verrelst, "Gaussian processes retrieval of crop traits in google earth engine based on sentinel-2 top-of-atmosphere data," *Remote Sensing of Environment*, vol. 273, p. 112 958, 2022, ISSN: 0034-4257. DOI: https://doi.org/10.1016/j.rse.2022.112958. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425722000724.

[12] J. Verrelst, J. Muñoz, L. Alonso, J. Delegido, J. P. Rivera, G. Camps-Valls, and J. Moreno, "Machine learning regression algorithms for biophysical parameter retrieval: Opportunities for sentinel-2 and -3," *Remote Sensing of Environment*, vol. 118, pp. 127–139, 2012, ISSN: 0034-4257. DOI: https://doi.org/10.1016/j.rse.2011.11.002. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S003442571100397X.

[13] L. von Rueden, S. Mayer, K. Beckh, B. Georgiev, S. Giesselbach, R. Heese, B. Kirsch, J. Pfrommer, A. Pick, R. Ramamurthy, M. Walczak, J. Garcke, C. Bauckhage, and J. Schuecker, "Informed machine learning – a taxonomy and survey of integrating prior knowledge into learning systems," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 1, pp. 614–633, 2023. DOI: 10.1109/TKDE.2021.3079836.

[14] L. von Rueden, S. Mayer, R. Sifa, C. Bauckhage, and J. Garcke, "Combining machine learning and simulation to a hybrid modelling approach: Current and future directions," in *Advances in Intelligent Data Analysis XVIII*, M. R. Berthold, A. Feelders, and G. Krempl, Eds., Cham: Springer International Publishing, 2020, pp. 548–560, ISBN: 978-3-030-44584-3.

[15] X. Gao, J. M. Gray, and B. J. Reich, "Long-term, medium spatial resolution annual land surface phenology with a bayesian hierarchical model," *Remote Sensing of Environment*, vol. 261, p. 112 484, 2021.

[16] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2014.

[17] C. Doersch, *Tutorial on variational autoencoders*, 2016. DOI: 10.48550/ARXIV.1606.05908. [Online]. Available: https://arxiv.org/abs/1606.05908.

[18] D. P. Kingma and M. Welling, "An introduction to variational autoencoders," *Foundations and Trends® in Machine Learning*, vol. 12, no. 4, pp. 307–392, 2019, ISSN: 1935-8237. DOI: 10.1561/2200000056. [Online]. Available: http://dx.doi.org/10.1561/2200000056.

[19] J. Lucas, G. Tucker, R. B. Grosse, and M. Norouzi, "Understanding posterior collapse in generative latent variable models," in *DGS@ICLR*, 2019.

[20] A. Makhzani and B. J. Frey, "Pixelgan autoencoders," in *NIPS*, 2017.

[21] Z. Cui, T. Gao, K. Talamadupula, and Q. Ji, "Knowledge-augmented deep learning and its applications: A survey," *ArXiv*, vol. abs/2212.00017, 2022.

[22] M. Raissi, P. Perdikaris, and G. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019, ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2018.10.045. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0021999118307125.

[23] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, "Physics-informed machine learning," *Nature Reviews Physics*, vol. 3, no. 6, pp. 422–440, 2021.

[24] Y. Yang, X. Zhang, Q. Guan, and Y. Lin, "Making invisible visible: Data-driven seismic inversion with spatio-temporally constrained data augmentation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–16, 2022. DOI: 10.1109/TGRS.2022.3144636.

[25] P. Y. Lu, S. Kim, and M. Solja či ć, "Extracting interpretable physical parameters from spatiotemporal systems using unsupervised learning," *Phys. Rev. X*, vol. 10, p. 031 056, 3 Oct. 2020. DOI: 10.1103/PhysRevX.10.031056. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevX.10.031056.

[26] Y. Su, J. Li, A. Plaza, A. Marinoni, P. Gamba, and S. Chakravortty, "DAEN: Deep autoencoder networks for hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 7, pp. 4309–4321, 2019. DOI: 10.1109/TGRS.2018.2890633.

[27] N. Takeishi and A. Kalousis, "Variational autoencoder with differentiable physics engine for human gait analysis and synthesis," in *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. [Online]. Available: https://openreview.net/forum?id=9ISlKio3Bt.

[28] M. A. Aragon-Calvo, "Self-supervised learning with physics-aware neural networks – i. galaxy model fitting," *Monthly Notices of the Royal Astronomical Society*, vol. 498, pp. 3713–3719, 2020.

[29] O. Rybkin, K. Daniilidis, and S. Levine, *Simple and effective vae training with calibrated decoders*, 2020.

[30] G. Dorta, S. Vicente, L. Agapito, N. D. Campbell, and I. Simpson, "Structured uncertainty prediction networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5477–5485.

[31] T. Koike-Akino and Y. Wang, "Autovae: Mismatched variational autoencoder with irregular posterior-prior pairing," in *2022 IEEE International Symposium on*

Information Theory (ISIT), 2022, pp. 1689–1694. DOI: 10.1109/ISIT50566.2022.9834769.

[32] A. Srivastava and C. Sutton, "Autoencoding variational inference for topic models," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, OpenReview.net, 2017. [Online]. Available: https://openreview.net/forum?id=BybtVK9lg.

[33] F. J. Kriegler, W. A. Malila, R. F. Nalepka, and W. Richardson, "Preprocessing Transformations and Their Effects on Multispectral Recognition," in *Remote Sensing of Environment, VI*, Jan. 1969, p. 97.

[34] W. Zhu, Y. Pan, H. He, L. Wang, M. Mou, and J. Liu, "A changing-weight filter method for reconstructing a high-quality NDVI time series to preserve the integrity of vegetation phenology," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 4, pp. 1085–1094, 2012. DOI: 10.1109/TGRS.2011.2166965.

[35] M. Hall-Beyer, "Comparison of single-year and multi-year NDVI time series principal components in cold temperate biomes," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 11, pp. 2568–2574, 2003. DOI: 10.1109/TGRS.2003.817274.

[36] E. F. Berra, R. Gaulton, and S. Barr, "Commercial off-the-shelf digital cameras on unmanned aerial vehicles for multitemporal monitoring of vegetation reflectance and NDVI," *IEEE transactions on geoscience and remote sensing*, vol. 55, no. 9, pp. 4878–4886, 2017.

[37] X. Yang, J. Mustard, J. Tang, and H. Xu, "Regional-scale phenology modeling based on meteorological records and remote sensing observations," *Journal of Geophysical Research*, vol. 117, 2012.

[38] X. Zhang, M. A. Friedl, C. B. Schaaf, A. H. Strahler, J. C. Hodges, F. Gao, B. C. Reed, and A. Huete, "Monitoring vegetation phenology using MODIS," *Remote Sensing of Environment*, vol. 84, no. 3, pp. 471–475, 2003.

[39] Y. Zérah, S. Valero, and J. Inglada, *Sentinel-2 time series for pheno-vae*, Nov. 2022. DOI: 10.5281/zenodo.7273500. [Online]. Available: https://doi.org/10.5281/zenodo.7273500.

[40] J. Inglada, A. Vincent, M. Arias, B. Tardy, D. Morin, and I. Rodes, "Operational High Resolution Land Cover Map Production at the Country Scale Using Satellite Image Time Series," *Remote Sensing*, vol. 9, no. 1, 2017, ISSN: 2072-4292. DOI: 10.3390/rs9010095. [Online]. Available: https://www.mdpi.com/2072-4292/9/1/95.

[41] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "Beta-VAE: Learning basic visual concepts with a constrained variational framework," in *International Conference on Learning Representations*, 2017. [Online]. Available: https://openreview.net/forum?id=Sy2fzU9gl.

[42] M. D. Homan and A. Gelman, "The no-u-turn sampler: Adaptively setting path lengths in hamiltonian monte carlo," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1593–1623, Jan. 2014, ISSN: 1532-4435.

[43] D. Phan, N. Pradhan, and M. Jankowiak, "Composable effects for flexible and accelerated probabilistic programming in numpyro," in *Program Transformations for ML Workshop at NeurIPS 2019*, 2019. [Online]. Available: https://openreview.net/forum?id=H1g1niFhIB.

[44] E. Bingham, J. P. Chen, M. Jankowiak, F. Obermeyer, N. Pradhan, T. Karaletsos, R. Singh, P. Szerlip, P. Horsfall, and N. D. Goodman, "Pyro: Deep universal probabilistic programming," *Journal of Machine Learning Research*, vol. 20, no. 28, pp. 1–6, 2019. [Online]. Available: http://jmlr.org/papers/v20/18-403.html.

[45] T. F. Coleman and Y. Li, "An interior trust region approach for nonlinear minimization subject to bounds," *SIAM Journal on Optimization*, vol. 6, no. 2, pp. 418–445, 1996.

[46] V. Edupuganti, M. Mardani, S. Vasanawala, and J. Pauly, "Uncertainty quantification in deep mri reconstruction," *IEEE Transactions on Medical Imaging*, vol. 40, no. 1, pp. 239–250, 2021. DOI: 10.1109/TMI.2020.3025065.

[47] M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U. R. Acharya, V. Makarenkov, and S. Nahavandi, "A review of uncertainty quantification in deep learning: Techniques, applications and challenges," *Information Fusion*, vol. 76, pp. 243–297, 2021, ISSN: 1566-2535. DOI: https://doi.org/10.1016/j.inffus.2021.05.008. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1566253521001081.

[48] R. Ak, V. Vitelli, and E. Zio, "An interval-valued neural network approach for uncertainty quantification in short-term wind speed prediction," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 11, pp. 2787–2800, 2015. DOI: 10.1109/TNNLS.2015.2396933.

[49] Z. Zheng, L. Wang, L. Yang, and Z. Zhang, "Generative probabilistic wind speed forecasting: A variational recurrent autoencoder based method," *IEEE Transactions on Power Systems*, vol. 37, no. 2, pp. 1386–1398, 2022. DOI: 10.1109/TPWRS.2021.3105101.

APPENDIX A
S2 DATA-SET

TABLE VIII
DISTRIBUTION OF THE LAND COVER CLASSES COMPOSING THE SENTINEL-2 TIME SERIES DATA-SET. THE CLASS LEGEND IS TAKEN FROM THE OSO [40] LAND COVER MAP PRODUCT.

Label	Percentage in data-set
Continuous Urban Fabric	0.6%
Discontinuous Urban Fabric	4.1%
Industrial and Commercial Units	3.1%
Road Surfaces	0.3%
Rapeseed	4.5%
Straw Cereals	9.9%
Protein Crops	2.5%
Soy	7.2%
Sunflower	33.0%
Corn	5.8%
Roots	0.2%
Intensive Grasslands	3.4%
Orchards	0.6%
Vineyards	1.8%
Broad-leaved Forests	6.7%
Coniferous Forests	5.5%
Grasslands	5.5%
Woody Moorlands	2.3%
Bare Rock	0.1%
Water Bodies	2.8%

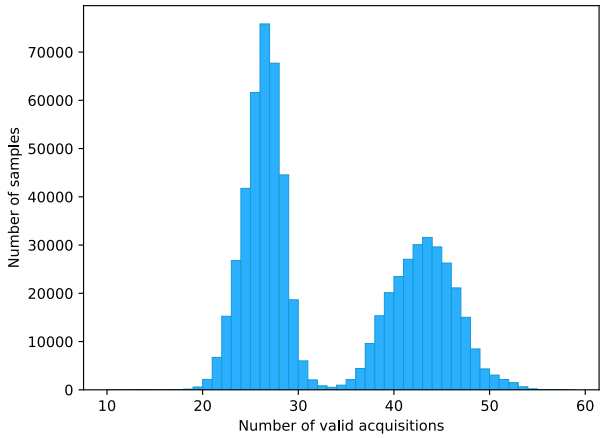


Fig. 12. Distribution of the temporal acquisitions composing the Sentinel-2 time series data-set.

APPENDIX B
RECONSTRUCTION OF S2 TIME SERIES

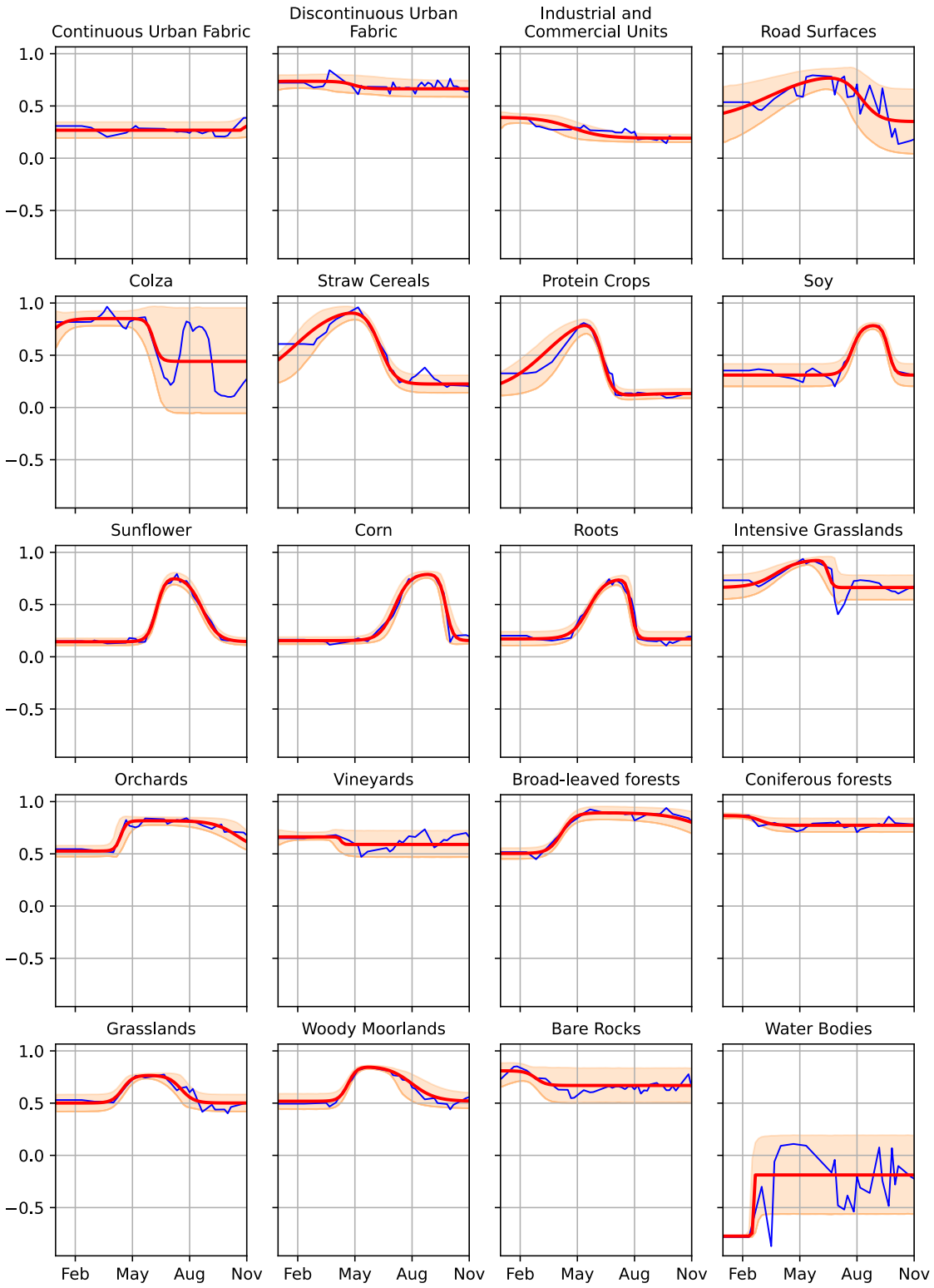


Fig. 13. Examples of reconstructions of Sentinel-2 NDVI time series with pheno-VAE trained on S2 data-set. Blue: 5-days interpolated S2 time series. Red: Reconstruction of the mode of phenological distribution. Orange: 5^{th}-95^{th} percentile interval.

APPENDIX C
INFERENCE PERFORMANCES

A. Metrics

In the following formulas for the metrics used in this article, $z^\star$ denotes a reference parameter, $\hat{z}$ the point estimate of the inferred distribution of a parameter, u and l are respectively the upper and lower bound of the inferred parameter distribution with a confidence level of $1-\alpha\%$. i denotes a sample number in the data-set of size N.

1) Mean Absolute Error:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|z_i^\star - \hat{z}_i\right|. \tag{22}$$

2) Mean Prediction Interval Width:

$$\text{MPIW}(\alpha) = \frac{\sum_{i=1}^{N} u_i(\alpha) - l_i(\alpha)}{N} \tag{23}$$

3) Prediction Interval Coverage Probability:

$$\text{PICP}(\alpha) = \frac{\#\left\{i \text{ s.t. } z_i^\star \in [l_i(\alpha), u_i(\alpha)]\right\}}{N} \tag{24}$$

B. Box-plots of the absolute error

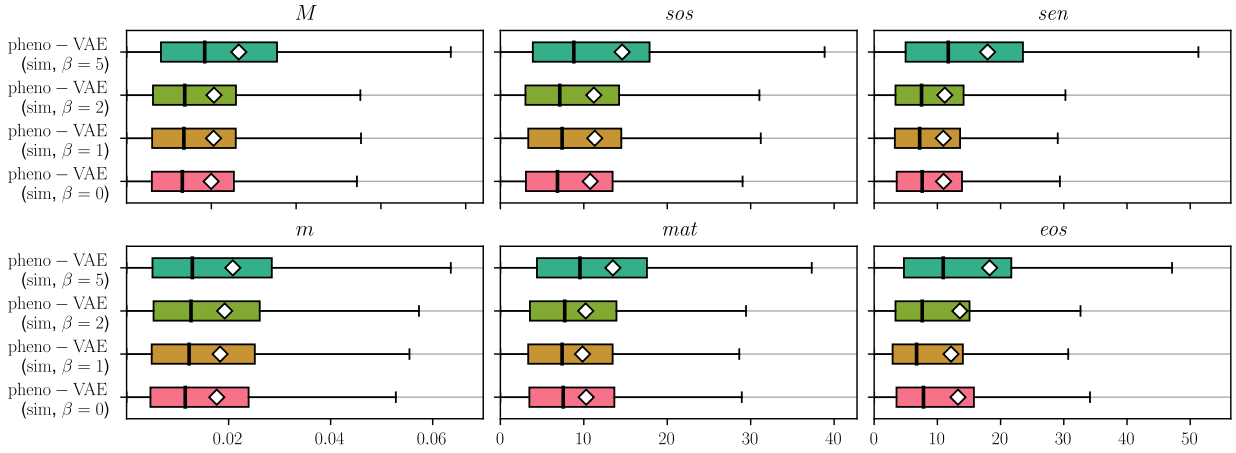


Fig. 14. Box-plot of the absolute error of inference of the 6 phenological parameters for pheno-VAE trained on S2 Data-set, with various settings of the coefficient β of the KL loss term. Box-plots are drawn from the results of the best fold of each method, in terms of the *eos* MAE. The white square for each box plot is the MAE. Absolute errors are comparable, except with $\beta = 5$, with a higher error.

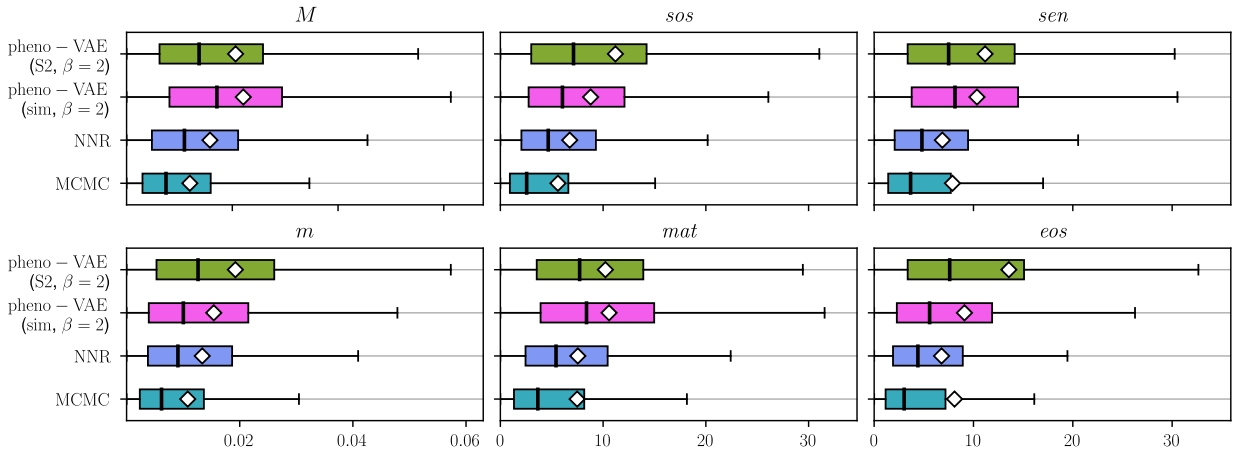


Fig. 15. Box-plot of the absolute error of inference of the 6 phenological parameters for MCMC, NNR, CF and pheno-VAE (with $\beta = 2$, trained on the S2 or simulated data-set). Box-plots are drawn from the results of the best fold of each method, in terms of the *eos* MAE. The white square for each box plot is the MAE. Absolute errors are the lowest for MCMC and Neural Network regression, and comparable for both pheno-VAE.

C. PICP as a function of the confidence level

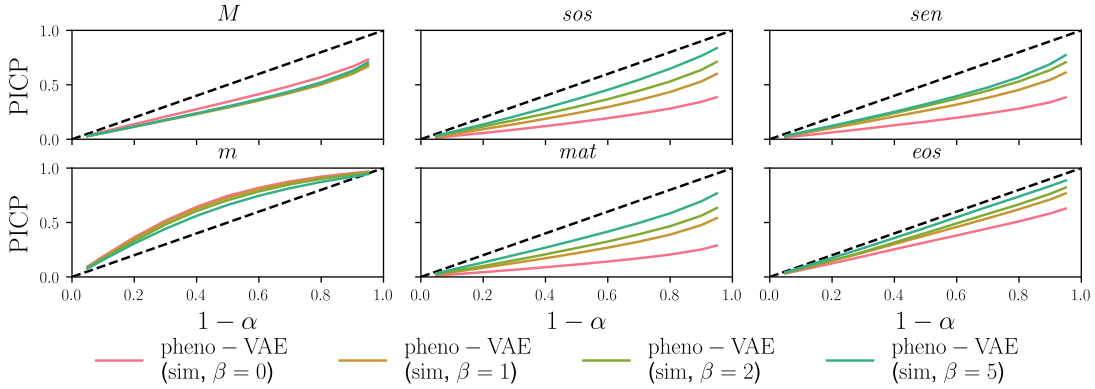


Fig. 16. PICP vs $1-\alpha$ for pheno-VAE trained on S2 Data-set, with various settings of the coefficient β of the KL loss term. The more β increases, the more the PICP increases at constant confidence level $1-\alpha$.

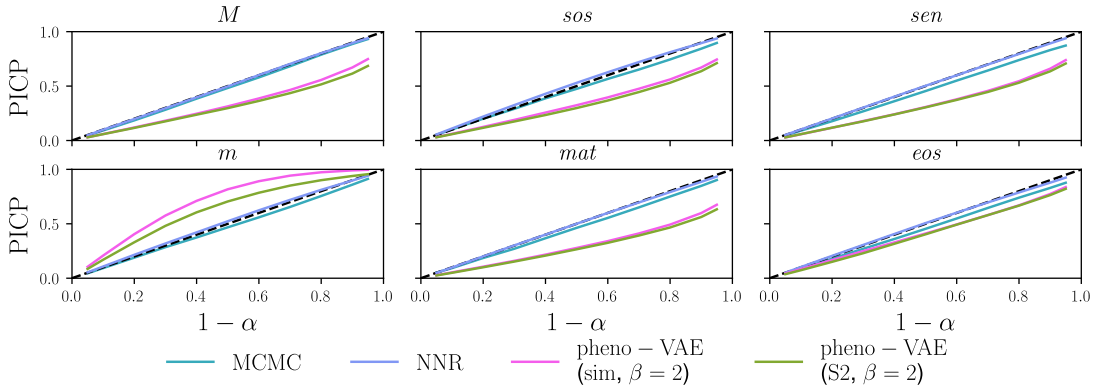


Fig. 17. PICP vs $1-\alpha$ for MCMC, Neural Network regression and pheno-VAE (with $\beta = 2$, trained on the S2 or simulated data-set). The PICP curves of Neural Network regression and MCMC are very close to PICP=α for all α, while pheno-VAE underestimates uncertainty for all confidence levels, for all phenological variables, except for m where uncertainty is overestimated.

D. MPIW as a function of the confidence level

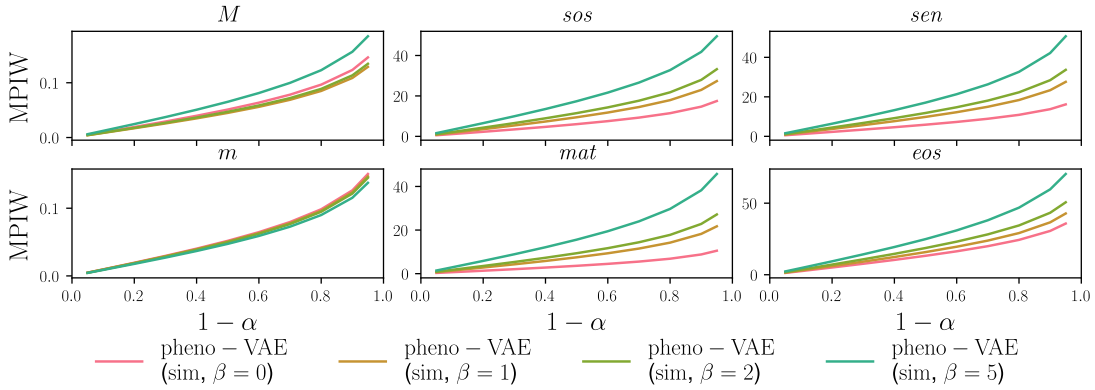


Fig. 18. MPIW vs $1-\alpha$ for pheno-VAE trained on S2 Data-set, with various settings of the coefficient β of the KL loss term. The more β increases, the more the MPIW increases at constant confidence level $1-\alpha$.

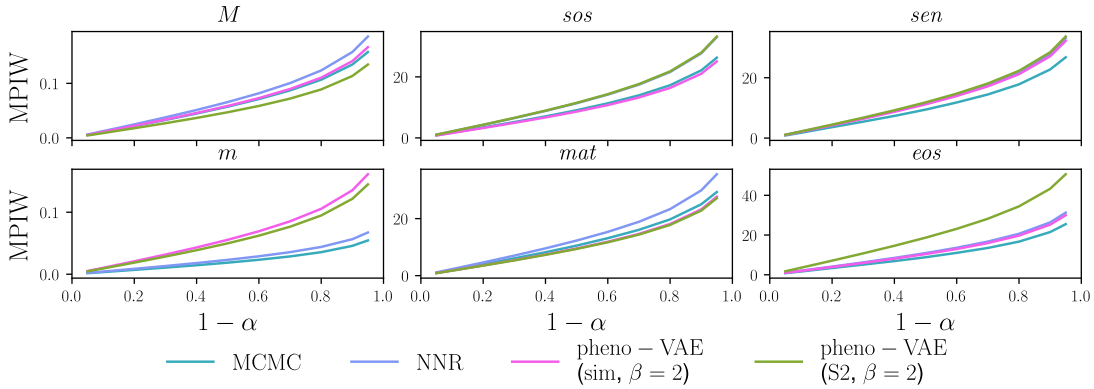


Fig. 19. MPIW vs $1-\alpha$ for MCMC, Neural Network regression and pheno-VAE (with $\beta = 2$, trained on the S2 or simulated data-set). prediction interval sizes are similar for all methods, except for m, where prediction intervals are larger for pheno-VAE, and for the eos of $pheno - VAE$ trained on the S2 data-set, that also has larger prediction intervals.

E. Box-plots of the prediction interval width

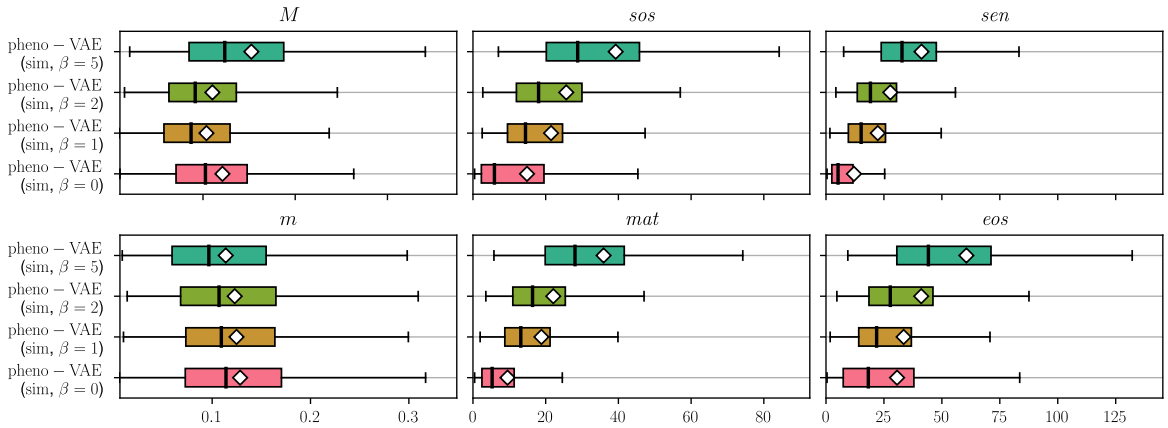


Fig. 20. Box-plot of the Prediction Interval Width (PIW) with a confidence level $1-\alpha = 0.90$ for pheno-VAE trained on S2 Data-set, with various settings of the coefficient β of the KL loss term. Box-plots are drawn fro the results of the best fold of each method, in terms of the *eos* MAE. The white square for each box plot is the MPIW. For phenological dates the PIW increases with β.

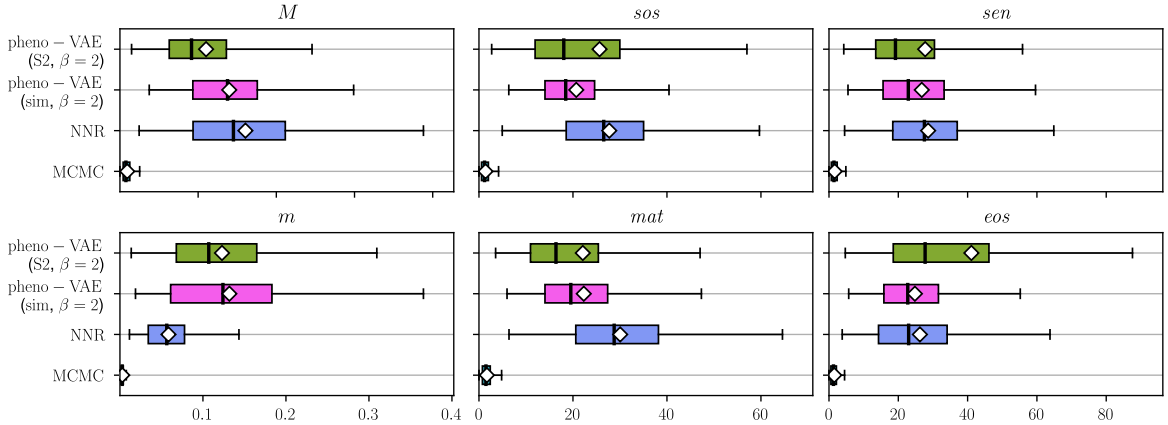


Fig. 21. Box-plot of the Prediction Interval Width (PIW) with a confidence level $1-\alpha = 0.90$ for the 6 phenological parameters for MCMC, Neural Network regression and pheno-VAE (with $\beta = 2$, trained on the S2 or simulated data-set). Box-plots are drawn from the results of the best fold of each method, in terms of the *eos* MAE. The white square for each box plot is the MPIW. MCMC infers significantly smaller PIW than the other methods that are comparable

APPENDIX D
DENSITY OF MAXIMUM OF CONTINUOUS DISTRIBUTIONS

Let Y be the maximum of n independent continuous random variables X_i. The CDF of Y is:

$$\begin{aligned} F_Y(y) &= P(Y < y) \\ &= P\left(\max_{i\in[[1,n]]} X_i < y\right) \\ &= P\left(\bigcap_{i=1}^{n} (X_i < y)\right) \\ &= \prod_{i=1}^{n} P\left(X_i < y\right) \\ &= \prod_{i=1}^{n} F_{X_i}(y) \end{aligned} \tag{25}$$

The log-derivative of the CDF of Y yields:

$$\begin{aligned} \frac{d\ln F_Y}{dy}(y) &= \frac{d}{dy}\ln\left(\prod_{i=1}^{n} F_{X_i}(y)\right) \\ &= \frac{d}{dy}\sum_{i=1}^{n}\ln\left(F_{X_i}(y)\right) \\ &= \sum_{i=1}^{n}\frac{d}{dy}\ln\left(F_{X_i}(y)\right) \\ &= \sum_{i=1}^{n}\frac{dF_{X_i}(y)}{dy}\frac{1}{F_{X_i}(y)} \\ &= \sum_{i=1}^{n} f_{X_i}(y)\frac{1}{F_{X_i}(y)} \end{aligned} \tag{26}$$

Finally, using the log-derivative of the CDF of Y enables deriving its PDF as a function of the PDFs and CDFs of X_i:

$$\begin{aligned} f_Y(y) &= \frac{dF_Y}{dy}(y) \\ &= F_Y(y)\frac{d\ln F_Y}{dy}(y) \\ &= \prod_{i=1}^{n} F_{X_i}(y)\sum_{i=1}^{n} f_{X_i}(y)\frac{1}{F_{X_i}(y)} \end{aligned} \tag{27}$$

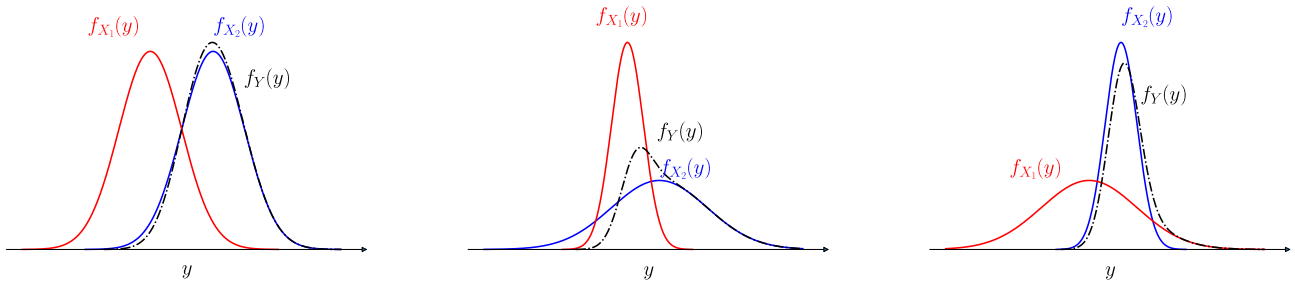


Fig. 22. Examples of distribution of the maximum Y of two Gaussian variables X_1 and X_2.

APPENDIX E
KL-DIVERGENCE OF TRUNCATED GAUSSIANS AND UNIFORM DISTRIBUTIONS

Let:

$$p \sim \mathcal{TN}\left(\mu, \sigma, a, b\right), \qquad q \sim \mathcal{U}\left(a, b\right) \tag{28}$$

with the truncated Gaussian density:

$$p(x) = \frac{\psi(\frac{x-\mu}{\sigma})}{\sigma\eta}, \qquad \psi(x) = \frac{e^{-\frac{x^2}{2}}}{\sqrt{2\pi}}$$

with

$$\eta = \Psi(\tilde{b}) - \Psi(\tilde{a}), \qquad \tilde{a} = \frac{a-\mu}{\sigma}, \qquad \tilde{b} = \frac{b-\mu}{\sigma}$$

and standard Gaussian CDF:

$$\Psi(x) = \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{x}{\sqrt{2}}\right)\right)$$

KL divergence is then:

$$\begin{aligned} \mathbb{KL}\left(p(x) \| q(x)\right) &= \int_a^b p(x) \ln \frac{p(x)}{q(x)} dx \\ &= \int_a^b p(x) \ln p(x) dx - \int_a^b p(x) \ln q(x) dx \end{aligned} \tag{29}$$

Its second term is:

$$\begin{aligned} \int_a^b p(x) \ln q(x) dx &= \int_a^b p(x) \ln \frac{1_{[a,b]}}{b-a} dx \\ &= -\ln\left(b-a\right) \int_a^b p(x) dx \\ &= -\ln\left(b-a\right) \end{aligned} \tag{30}$$

The first term is:

$$\begin{aligned} \int_a^b p(x) \ln p(x) dx &= \int_a^b p(x) \ln \frac{\psi(\frac{x-\mu}{\sigma})}{\sigma\eta} dx \\ &= -\ln\left(\sigma\eta\right) \int_a^b p(x) dx + \int_a^b p(x) \ln \psi\left(\frac{x-\mu}{\sigma}\right) dx \\ &= -\ln\left(\sigma\eta\right) + \int_a^b p(x) \ln \frac{e^{-\frac{(x-\mu)^2}{2\sigma^2}}}{\sqrt{2\pi}} dx \\ &= -\ln\left(\sigma\eta\right) - \frac{1}{2}\ln\left(2\pi\right) - \int_a^b p(x) \frac{\left(x-\mu\right)^2}{2\sigma^2} dx \\ &= -\ln\left(\sigma\eta\right) - \frac{1}{2}\ln\left(2\pi\right) - \frac{1}{2\sigma^2} \int_a^b p(x) \left(x^2 - 2\mu x + \mu^2\right) dx \\ &= -\ln\left(\sigma\eta\right) - \frac{1}{2}\ln\left(2\pi\right) - \frac{\mu^2}{2\sigma^2} - \frac{1}{2\sigma^2} \int_a^b x^2 p(x) dx + \frac{\mu}{\sigma^2} \int_a^b x p(x) dx \\ &= -\ln\left(\sigma\eta\right) - \frac{1}{2}\ln\left(2\pi\right) - \frac{\mu^2}{2\sigma^2} - \frac{1}{2\sigma^2}\left\langle p^2 \right\rangle + \frac{\mu}{\sigma^2}\left\langle p \right\rangle \end{aligned} \tag{31}$$

with truncated Gaussian moments:

$$\left\langle p^2 \right\rangle = \sigma^2 + \frac{\sigma^2}{\eta}\left(\tilde{a}\psi(\tilde{a}) - \tilde{b}\psi(\tilde{b})\right) + \mu^2 + \frac{2\mu\sigma}{\eta}\left(\psi(\tilde{a}) - \psi(\tilde{b})\right)$$

$$\left\langle p \right\rangle = \mu + \frac{\sigma}{\eta}\left(\psi(\tilde{a}) - \psi(\tilde{b})\right)$$

Finally:

$$\mathbb{KL}\left(p(x) \| q(x)\right) = -\frac{1}{2} - \frac{1}{2}\ln\left(2\pi\right) - \ln\left(\sigma\eta\right) - \frac{\tilde{a}\psi(\tilde{a}) - \tilde{b}\psi(\beta)}{2\eta} + \ln(b-a)$$

APPENDIX F
NOTATIONS

A. Variables notations

Bold font denotes a vector or a matrix variable. Variables with a hat denote an estimated quantity. Underlined variables are rectified variables. Indexing with i denotes a dimension of latent variables, and indexing with j denotes an element of a data-set.

Notation	Definition
$\mathbf{x}$	Observation, input data
$\hat{\mathbf{x}}$	Reconstruction of input data
$\mathbf{z}$	Latent variable
$\underline{z}$	Rectified latent variable
$\boldsymbol{\lambda}$	Parameter of variational distribution
ϕ	Parameters of encoder's neural network
θ	Parameters of decoder's neural network
$1-\alpha$	Confidence level
β	Coefficient on the KL term in the ELBO used in β-VAE
$\boldsymbol{\mu}_{\mathbf{z}}$	Mean parameter of Gaussian latent space
$\underline{\mu}_{z_i}$	Rectified mean of Gaussian latent distribution
$\boldsymbol{\Sigma}_{\mathbf{z}}$	Covariance matrix of Gaussian latent space
$\boldsymbol{\mu}_{\hat{\mathbf{x}}}$	Mean parameter of Gaussian decoder distribution
$\boldsymbol{\Sigma}_{\hat{\mathbf{x}}}$	Covariance matrix of Gaussian decoder distribution
K	Number of latent samples drawn to estimate the decoder's output distribution parameters
Δ_z	difference between two consecutive latent variables
ρ	Reflectance
$\mathcal{F}$	General notation for user defined decoder
$\Omega_{\mathbf{z}}$	Double-sigmoid function parametrized by $\mathbf{z}$
a, b	Bounds of a variational distribution support
l, u	prediction interval bounds
N	Number of samples in test data-set
S	Sigmoid function
ψ	Gaussian PDF
Ψ	Gaussian CDF
$\mathcal{U}$	Uniform distribution
$\mathcal{N}$	Gaussian distribution
$\mathcal{TN}$	Truncated Gaussian distribution
$\mathbb{KL}$	Kullback-Leibler divergence
$\mathbb{E}$	Expectation
$\#$	Cardinality
$\emptyset$	Empty set

B. Acronyms

Acronym	Definition
ELBO	Evidence Lower BOund
KL	Kullback-Leibler (Divergence)
VAE	Variational Autoencoder
NLL	Negative Log-Likelihood
NNR	Neural Network Regression
MCMC	Markov Chain Monte Carlo
CF	Curve-Fitting
PDF	Probability Density Function
CDF	Cumulative Distribution Function
NDVI	Normalized Difference Vegetation Index
MAE	Mean Absolute Error
MPIW	Mean Prediction Interval Width
PICP	Prediction Interval Coverage Probability
CES OSO	Centre d'Expertise Scientifique sur l'Occupation des Sols