

---

# MAGIC: Modular Auto-encoder for Generalisable Model Inversion with Bias Corrections

---

**Yihang She**

University of Cambridge  
ys611@cam.ac.uk

**Clement Atzberger**

Mantle Labs  
clement@mantle-labs.com

**Andrew Blake**

University of Cambridge  
ab@ablake.ai

**Adriano Gualandi**

University of Cambridge  
ag2347@cam.ac.uk

**Srinivasan Keshav**

University of Cambridge  
sk818@cam.ac.uk

## Abstract

Scientists often model physical processes to understand the natural world and uncover the causation behind observations. Due to unavoidable simplification, discrepancies often arise between model predictions and actual observations, in the form of systematic biases, whose impact varies with model completeness. Classical model inversion methods such as Bayesian inference or regressive neural networks tend either to overlook biases or make assumptions about their nature during data preprocessing, potentially leading to implausible results. Inspired by recent work in inverse graphics, we replace the decoder stage of a standard autoencoder with a physical model followed by a bias-correction layer. This generalisable approach simultaneously inverts the model and corrects its biases in an end-to-end manner without making strong assumptions about the nature of the biases. We demonstrate the effectiveness of our approach using two physical models from disparate domains: a complex radiative transfer model from remote sensing; and a volcanic deformation model from geodesy. Our method matches or surpasses results from classical approaches without requiring biases to be explicitly filtered out, suggesting an effective pathway for understanding the causation of various physical processes. The code is available on <https://github.com/yihshe/MAGIC.git>.

## 1 Introduction

The quantitative and mathematical modeling of natural phenomena is a foundational task in science. Inverting a physical model, to deduce underlying causal variables from observations, is a powerful technique that has been used widely [Adler and Holder, 2021, Lahoz and Schneider, 2014, Zhdanov, 2015]. Prior attempts at model inversion have struggled with the issue of model incompleteness, where outcomes predicted by a model differ from actual outcomes [Groetsch and Groetsch, 1993, Zhdanov, 2015]. These approaches either overlook model biases or make assumptions about them during data preprocessing, leading to significant error in inferred variable values.

Two major recent advances encourage us to revisit this long-standing problem. First, modern sensor technologies have made available extensive data-sets of measured observations [Claverie et al., 2018]. Second, although standard statistical approaches in machine learning may be unable to yield interpretable representations of that data [Chen et al., 2016], in recent years auto-encoders have been used to uncover disentangled representations in low dimensions [Locatello et al., 2019]. However these approaches lack an effective inference mechanism to obtain variables with clearly defined physical meanings [Kumar et al., 2018].

It is notable that researchers in graphics have explored incorporating a differentiable renderer into an auto-encoder to recover interpretable representations from visual data [Loper and Black, 2014, Kato and Harada, 2019]. Similar approaches have also been used in some specific physics problems [Lu et al., 2020, Zérah et al., 2022] but it is unclear whether these approaches can be generalized. This is because auto-encoder gradients need to backpropagate through the physical model but existing physical models are often written in legacy code, and not designed to be differentiable in the way that deep learning demands.

A further issue is that existing autocoder-based approaches rarely compensate for systematic biases in physical models. Ideally, a physical model’s outputs exhibit no systematic bias when compared with measured observations. However, real-world models may exhibit particularly prominent systematic biases because they only approximately represent a complex, and often noisy, physical system [Abramowitz et al., 2007, Zhdanov, 2015, Widłowski et al., 2013]. This requires bias correction, with the complexity of the bias correction depending on the incompleteness of the physical model. Unfortunately, the addition of a bias correction layer can modify the causal relationship between the physical model’s causal variables and the corrected output. Thus, there is a fine balance between the complexity of bias correction and interpretability of the causal variables extracted by inversion of the bias corrected model.

To address these issues, we first demonstrate how to augment a deterministic physical model with a bias correction layer that learns a transformation from model output to observed outcomes. This greatly increases model prediction accuracy. Second, we use an autoencoder-like network to invert this augmented model to extract causal variables from observations. We do this in two disparate physical science domains: a radiative transfer model (RTM) [Atzberger, 2000] that computes spectral signatures from biophysical variables and is not inherently differentiable; and a volcanic deformation model [Mogi, 1958] that computes surface deformations due to magma chamber activity. We demonstrate the generalisability of our approach by making the complex RTM fully differentiable and stable during training, using the power of Large Language Models (LLMs), an approach that could be applied more generally to inverting legacy implementations of complex physical models. Finally, we use two physical models with different levels of completeness to investigate the complexity of the corresponding bias correction.

By detailed comparison of learning outcomes against baselines derived from classical methods, we demonstrate that auto-encoders can be used as a generalizable approach for simultaneous model inversion and bias correction. This suggests a pathway for a more accurate understanding of the causation of diverse physical processes, by means of end-to-end learning.

## 2 Related Work

### 2.1 Disentangled representation learning

A key aspect of our approach is on interpretability and disentanglement. This relates to the concept of disentangled representation learning, which seeks to uncover lower-dimensional, semantically meaningful factors of variation from high-dimensional data [Locatello et al., 2019]. In the paradigm of unsupervised learning of disentangled representations, Variational Autoencoders (VAEs) [Kingma and Welling, 2013] and their variants, such as  $\beta$ -VAE [Higgins et al., 2017], are based on probabilistic modeling. These approaches allow disentangled variables to emerge from learning [Kumar et al., 2018]. Alternatively, Generative Adversarial Networks (GANs) [Goodfellow et al., 2014] have also demonstrated success in learning interpretable and disentangled representations for image synthesis, as seen in InfoGAN [Chen et al., 2016] and StyleGAN [Karras et al., 2019]. However, as generative models, the disentangled variables typically emerge only during learning and lack an effective inference mechanism [Kumar et al., 2018]. Moreover, as Locatello et al. [2019] suggested, inductive biases are essential for unsupervised approaches to learning disentangled representations. For physical models, the situation is simplified because the forward model is deterministic and benefits from having a predetermined set of interpretable variables, namely the physical input parameters to the model.

## 2.2 Inverse problems in physical sciences

Across various domains in natural sciences, forward numerical models are developed based on physical principles and inverting them has led to significant applications e.g. electrical impedance tomography in healthcare [Adler and Holder, 2021], data assimilation in climate science [Lahoz and Schneider, 2014], and understanding the Earth’s structure in geophysics [Zhdanov, 2015]. We test our approach on two domains (see section 3.2). First, Radiative Transfer Models (RTMs) from remote sensing [Suits, 1971, Atzberger, 2000] belong to a family of canopy reflectance models that simulate spectral bidirectional forest reflectance, given inputs of important forest characteristics [Widlowski et al., 2013, Li and Strahler, 1985, Suits, 1971, Atzberger, 2000]. Second, volcanic deformation models from geodesy simulate surface deformation due to magma chamber activity [Segall, 2013, Mogi, 1958] and come from a broader family of geophysical models where inversion is a key problem [Zhdanov, 2015]. Classical inversion approaches include Bayesian inference, numerical optimization, and look-up tables [Groetsch and Groetsch, 1993, Goel, 1988, Combal et al., 2003]. More recently, neural network regressors have also been used [Gong, 1999, Schlerf and Atzberger, 2006, Mousavi et al., 2024].

## 2.3 Autoencoder as an approach to inversion

We do not claim originality simply for using auto-encoders to invert physical models; that has been extensively investigated in computer science, especially in inverse graphics [Kato et al., 2020]. A seminal contribution is OpenDR [Loper and Black, 2014], which proposes differentiable rendering to invert a graphics renderer by learning with backpropagation. Since then, auto-encoders have been a popular choice to incorporate a differentiable renderer into self-supervised learning pipelines [Kato et al., 2020], and have been applied to 3D reconstruction, and human pose estimation from single-view images [Kato and Harada, 2019, Pavlakos et al., 2018]. This approach has been applied to a few scientific domains [Lu et al., 2020, Zérah et al., 2022], and recently, specifically for a canopy-reflectance model [Zérah et al., 2023]. However none of these studies addresses systematic bias, and that is what is especially distinctive about our approach.

## 3 Methods

### 3.1 Learning the inverse of physical models

To infer physical variables  $Z$  from measured observations  $X$ , we use an auto-encoder architecture with bias correction to train an encoder that maps measured observations to physical variables (Fig. 1). The physical process that generates the observations  $X$  is written:

$$X = \mathbf{F}(Z) + \mathbf{B}(Q) + \eta \quad (1)$$

where  $\mathbf{F}$  is the modelled process acting on physical variables  $Z$ ;  $\mathbf{B}(Q)$  is the systematic bias between  $\mathbf{F}(Z)$  and  $X$  resulting from unmodeled processes  $\mathbf{B}$ , and depending on a collection of other variables  $Q$ ; and  $\eta$  represents random noise.

A classical auto-encoder  $\mathbf{M}_A$  consists of an encoder  $\mathbf{E}_A$  and a decoder  $\mathbf{D}_A$  (eq. (2)).

$$X_A = \mathbf{D}_A(\mathbf{E}_A(X)) \quad (2)$$

Given a sufficiently sophisticated network architecture,  $\mathbf{M}_A$  can minimize Mean Squared Error  $\mathcal{L}_{\text{MSE}}(\theta_{\mathbf{E}_A})$  between  $X$  and  $X_A$ , but the latent codes  $Z_A$  computed as  $\mathbf{E}_A(X)$  need not be semantically interpretable in terms of well defined physical properties. Replacing  $\mathbf{D}_A$  in eq. (2) with the physical model  $\mathbf{F}$  however ensures that an encoder  $\mathbf{E}_B$  embeds  $X$  as interpretable physical variables  $Z_B = \mathbf{E}_B(X)$ , which are then passed to  $\mathbf{F}$  to reconstruct observations (autoencoder  $\mathbf{M}_B$ ):

$$X_B = \mathbf{F}(\mathbf{E}_B(X)) \quad (3)$$

Ideally, when  $\mathbf{F}$  is perfectly built to account fully for the physical processes, there is no need for  $\mathbf{B}(Q)$  in eq. (1). In that case,  $\mathcal{L}_{\text{MSE}}(\theta_{\mathbf{E}_B})$  should be on a par with  $\mathcal{L}_{\text{MSE}}(\theta_{\mathbf{E}_A})$ . In reality the rigidity and oversimplification of  $\mathbf{F}$  gives appreciable reconstruction mismatch between  $X$  and  $X_B$  so we add additional non-linear bias-correction layers  $\mathbf{C}$  after  $\mathbf{F}$  (autoencoder  $\mathbf{M}_C$ , also see Fig. 1):

$$X_C = \mathbf{C}(\mathbf{F}(\mathbf{E}_C(X))). \quad (4)$$

The correction layer  $\mathbf{C}$  increases the flexibility of  $\mathbf{F}$  and decreases  $\mathcal{L}_{\text{MSE}}(\theta_{\mathbf{E}_C}, \theta_C)$  to improve the learning outcome of  $Z_C$ . The risk, though, is that input variables  $Z_C$  are no longer anchored as strongly, especially if with appreciable flexibility in  $\mathbf{C}$ , allowing  $Z_C$  to entangle and drift away from physical interpretations. A solution is to limit the complexity of  $\mathbf{C}$  to two linear transformations with a single ReLU activation  $\sigma$  between them:

$$X_C = W_2 \sigma(W_1 \mathbf{F}(\mathbf{E}_C(X)) + b_1) + b_2. \quad (5)$$

Here,  $W_1$  expands the input to four times its original dimension, and  $W_2$  reduces it back to its original size. This design mirrors the feed-forward head in Transformers [Vaswani et al., 2017], which has allowed just enough flexibility in non-linear transformations — exactly what we need (see section 4).

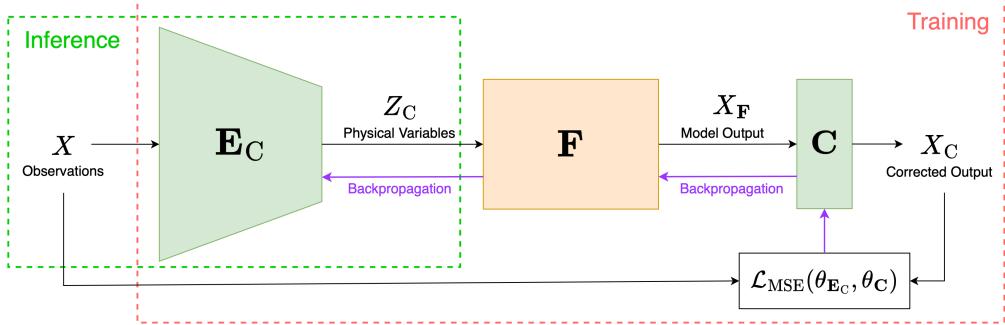


Figure 1: **Learning the inverse**, end-to-end, in an autoencoder that includes correction layers  $\mathbf{C}$ .

### 3.2 Physical models for inversion

**INFORM:** The INFORM radiative transfer model from remote sensing (denoted by  $\mathbf{F}_{\text{RTM}}$ ), developed by Atzberger [2000], consists of a set of sub-models designed to simulate the reflectance of forest canopies given parameters representing their biophysical and geometric properties. Despite decades of research, the complexity of forest structures leads to discrepancies between simulated and measured spectra. These biases stem from over-simplification of the physics and significantly affect accuracy of inferred variables with classical approaches [Widlowski et al., 2013]. We aim to estimate certain set of the biophysical variables, whose plausible ranges are known (Tab. 1), while fixing other variables at typical values for the study area (see Appendix A.3). In practice,  $\mathbf{F}_{\text{RTM}}$  is implemented with NumPy and comprises approximately 1700 lines of code, and is not explicitly differentiable.

**Mogi:** The Mogi model (denoted by  $\mathbf{F}_{\text{Mogi}}$ ) is a model used in geodesy to describe the displacement field at the surface of the Earth resulting from a spherical pressure source, typically a magma chamber, at depth [Mogi, 1958]. The inverse problem for  $\mathbf{F}_{\text{Mogi}}$  involves estimating the location and volume change of magma chamber activity from observed surface displacements (Tab. 2 and Appendix A.5). Compared with  $\mathbf{F}_{\text{RTM}}$ ,  $\mathbf{F}_{\text{Mogi}}$  is built on simple assumptions, with the complexity of the Earth's crust being crudely approximated by an elastic half-space. Despite its simplicity, the Mogi model is able to capture the first-order patterns of surface deformation associated with volcanic sources. The challenge is that, the volcanic deformation can be small if compared to other sources of deformation (e.g. hydrological loading), posing challenges to the bias correction (see Appendix A.6.2).

Table 1: **Biophysical variables of  $\mathbf{F}_{\text{RTM}}$  to learn.** Bound ranges of  $Z_h$  and  $Z_{cd}$  are marked with \*, as they will be inferred from  $Z_{fc}$  and fixed stem density using allometric equations [Jucker et al., 2017] to mitigate the ill-posed nature of the inverse problem (see Appendix A.3).

		Leaf Model				Canopy Model		Forest Model		
Sub-model	Variable	Structure Parameter	Chlorophyll A+B	Water Content	Dry Matter	Leaf Area Index	Undergrowth LAI	Fractional Coverage	Tree Height	Crown Diameter
Acronym	N	cab	cw	cm	LAI	LAIu	fc	h	cd	
Min	1	10	0.001	0.005	0.01	0.01	0.1	*	*	
Max	3	80	0.02	0.05	5	1	1	*	*	

Table 2: **Geophysical variables of  $\mathbf{F}_{\text{Mogi}}$  to learn.** The source center ( $x_m, y_m$ ) is assumed to lie within the GNSS stations’ boundaries (section 3.4), with depth and volume ranges matching those used in [Walwer et al., 2016].

Variable	Epicenter-X (km)	Epicenter-Y (km)	Depth (km)	Volume Change ( $10^6 \text{ m}^3$ )
Acronym	$x_m$	$y_m$	$d$	$\Delta V$
Min	-9.33	-5.80	2	-10
Max	14.35	7.62	20	10

### 3.3 End-to-end learning with physical models

Differentiability is required for auto-encoder workflow and the implementation of  $\mathbf{F}_{\text{Mogi}}$  in PyTorch is straightforward as it only involves a few lines of equations (Appendix A.5). The RTM  $\mathbf{F}_{\text{RTM}}$  is complex and all its NumPy operations need conversion to PyTorch, identifying functions that are not differentiable, and constructing backward function for each to obtain approximated derivatives.

Recent developments in LLMs would suggest using GPT-4 [OpenAI, 2023] so that common operations get automatically converted to PyTorch, and others are flagged for custom treatment. In this way, we converted 1,742 lines of code in less than a week (see Appendix A.1). In practice, canopy-radiation interaction needs exponential, logarithm, and square root functions which can lead to numerically unstable derivatives. This is handled by bypassing instability points during backpropagation (see Appendix A.2).

### 3.4 Datasets

The dataset for  $\mathbf{F}_{\text{RTM}}$  consists of 17692 individual spectra  $X_{S2} = \{x_{S2,i}, i = 1, \dots, 11\}$ , extracted from 1283 individual sites across Austria observed by Sentinel-2 [Drusch et al., 2012]. Spectra  $X_{S2}$  form a temporal sequence covering April to October 2018 — both coniferous and deciduous forests — consisting of 12 species. Spectra comprise 11 bands (see Appendix A.4).

For  $\mathbf{F}_{\text{Mogi}}$ , the dataset comes from 12 Global Navigation Satellite System (GNSS) stations around the Akutan Volcano, one of the most active volcanoes in the Aleutian Islands (Alaska, USA). Every day, each GNSS station records its position at the surface in east, north and vertical directions (denoted  $X_{\text{GNSS}} = \{x_{e,i}, x_{n,i}, x_{v,i}, i = 1, \dots, 12\}$ ). In total, we have observations from 6525 consecutive days from 2006 until early 2024 (<http://geodesy.unr.edu>, [Blewitt et al., 2018]) (see Appendix A.6).

For both problems, collecting ground truth data is largely impractical. The interpretation of results has to rely on plausibility checks on variable distributions and temporal evolution patterns. For  $\mathbf{F}_{\text{RTM}}$ , we use prior knowledge of tree species and temporal data to validate biophysical estimates. For  $\mathbf{F}_{\text{Mogi}}$ , geophysical estimates of a magma source and reconstructed transient GNSS signals are validated against existing literature [Ji and Herring, 2011, Walwer et al., 2016].

### 3.5 Baseline

In addition to comparing  $\mathbf{M}_A$ ,  $\mathbf{M}_B$ , and  $\mathbf{M}_C$ , we have also compared with classical approach baselines. Further implementation details can be found in Appendix A.8.

For  $\mathbf{F}_{\text{RTM}}$ , we trained a neural network regressor  $\mathbf{M}_{D,\text{RTM}}$  using sampled variables and spectra from  $\mathbf{F}_{\text{RTM}}$  — a classical approach for inversion [Gong, 1999] as a baseline. Model  $\mathbf{M}_{D,\text{RTM}}$  shares the same architecture as the encoder of  $\mathbf{M}_{C,\text{RTM}}$ . It is first trained to regress the sampled variables from the synthetic spectra and then applied to the real spectra  $X_{S2}$  to predict physical variables ( $Z_{D,\text{RTM}}$ ). To obtain the spectral reconstruction  $X_{D,\text{RTM}}$ ,  $Z_{D,\text{RTM}}$  is passed to  $\mathbf{F}_{\text{RTM}}$ .

For  $\mathbf{F}_{\text{Mogi}}$ , we compared our inversion results with classical results [Walwer et al., 2016] which preprocessed the GNSS time series over 2006-2014 as follows. First they used Singular Spectrum Analysis to detect likely transient signals by inspecting the processed time series across stations and directions. Then, they performed a grid search of parameters to invert  $\mathbf{F}_{\text{Mogi}}$  by fitting its estimates to the extracted transient signals. They fixed the location of the Mogi source and estimated the depth and volume change for periods likely due to magma movement. Notably they detected significant

volcanic inflation of Akutan in early 2008, also reported by Ji and Herring [2011]. Thus for validation, we have saved the GNSS time series from 2006 to 2009 as a test set to cover the 2008 event.

## 4 Results

### 4.1 Effects of bias correction

The systematic bias of  $\mathbf{F}_{\text{RTM}}$  is evident comparing the input spectra  $X_{S2}$  with the reconstruction  $X_{S2,B}$  from  $\mathbf{M}_{B,\text{RTM}}$  trained without bias correction (Fig. 2a). Further illustrations of bias corrections are detailed in Appendix A.9.

Similarly  $\mathbf{F}_{\text{Mogi}}$  shows systematic bias comparing  $X_{\text{GNSS}}$  with the reconstruction from  $\mathbf{M}_{B,\text{Mogi}}$  (Fig. 2b). Bias correction using  $\mathbf{M}_{C,\text{Mogi}}$  achieves some improvement in reconstruction of GNSS signals (Tab. 4) though less dramatically than  $\mathbf{M}_{C,\text{RTM}}$ , as indicated by the  $R^2$  values (Figs. 2a and 2b). The RTM is a more complex model, presenting greater challenges in inversion, but imposes a strong prior and captures physical processes effectively. In contrast,  $\mathbf{F}_{\text{Mogi}}$  makes simpler assumptions — considering only a point-like pressure source for surface deformation — forming a less informative bottleneck in the decoder and so poses a greater challenge for bias correction. However our correction layer is still able to capture most of the systematic bias and improves the learned inverse. Bias

Table 3: **RTM: The reconstruction loss of models under different steps of the inversion.**  $\mathcal{L}_{\text{MSE}}$  under  $\mathbf{M}_{C,\text{RTM}}$  is comparable to the classical auto-encoder  $\mathbf{M}_{A,\text{RTM}}$ , and is reduced by more than an order of magnitude compared to the regressor baseline  $\mathbf{M}_{D,\text{RTM}}$ .

Model	Ablation	$\mathcal{L}_{\text{MSE},\text{train}}$	$\mathcal{L}_{\text{MSE},\text{val}}$	$\mathcal{L}_{\text{MSE},\text{test}}$
$\mathbf{M}_{A,\text{RTM}}$	w/o $\mathbf{F}_{\text{RTM}}$ , w/o $\mathbf{C}_{\text{RTM}}$	0.0193	0.0219	0.0191
$\mathbf{M}_{B,\text{RTM}}$	w/ $\mathbf{F}_{\text{RTM}}$ , w/o $\mathbf{C}_{\text{RTM}}$	0.0875	0.0833	0.0856
$\mathbf{M}_{C,\text{RTM}}$	w/ $\mathbf{F}_{\text{RTM}}$ , w/ $\mathbf{C}_{\text{RTM}}$	0.0210	0.0235	0.0217
$\mathbf{M}_{D,\text{RTM}}$	N/A	-	-	0.6676

Table 4: **Mogi: The reconstruction loss under different steps.**  $\mathcal{L}_{\text{MSE}}$  for  $\mathbf{M}_{C,\text{Mogi}}$  outperforms  $\mathbf{M}_{B,\text{Mogi}}$  and matches  $\mathbf{M}_{D,\text{Mogi}}$ , but the bias correction is less pronounced compared to the RTM model — a more complete model with less complex biases to address (see Fig. 2).

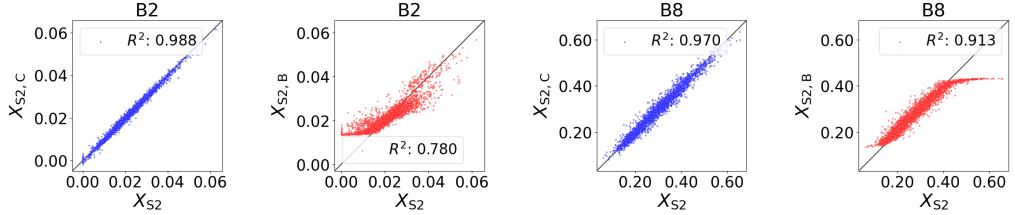
Model	Ablation	$\mathcal{L}_{\text{MSE},\text{train}}$	$\mathcal{L}_{\text{MSE},\text{val}}$	$\mathcal{L}_{\text{MSE},\text{test}}$
$\mathbf{M}_{A,\text{Mogi}}$	w/o $\mathbf{F}_{\text{Mogi}}$ , w/o $\mathbf{C}_{\text{Mogi}}$	0.4389	0.9119	0.6324
$\mathbf{M}_{B,\text{Mogi}}$	w/ $\mathbf{F}_{\text{Mogi}}$ , w/o $\mathbf{C}_{\text{Mogi}}$	0.6998	0.9721	0.8095
$\mathbf{M}_{C,\text{Mogi}}$	w/ $\mathbf{F}_{\text{Mogi}}$ , w/ $\mathbf{C}_{\text{Mogi}}$	0.4803	0.9338	0.7110

correction also enhances the learning of variables in latent space for  $\mathbf{F}_{\text{RTM}}$  and for  $\mathbf{F}_{\text{Mogi}}$  — compare retrieved variables with and without a bias correction (Fig. 3) — for  $\mathbf{F}_{\text{RTM}}$  the distributions of several variables are distorted to extremes if bias is not corrected (Fig. 3a).

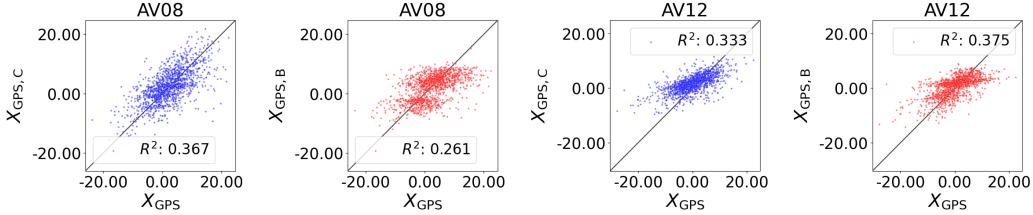
Similarly for  $\mathbf{F}_{\text{Mogi}}$  bias pushes the center of the Mogi source to extremes (Fig. 3b) if not corrected, and although  $Z_d$  and  $Z_{\Delta V}$  follow unimodal or bimodal distributions, they have implausible value ranges without bias correction. Both Ji and Herring [2011] and Walwer et al. [2016] have suggested that the depth  $Z_d$  less than 10 km and the volume change  $Z_{\Delta V}$  range rarely exceed  $-5 \cdot 10^6 \text{ m}^3$  to  $5 \cdot 10^6 \text{ m}^3$  for the evaluated period at the Akutan volcano [Walwer et al., 2016]. Indeed, if  $Z_d$  exceeds 10 km, then the volume change required to reproduce the observed displacement would be unreasonably large. It is possible that, without bias correction, the model attempts to replicate a common mode signal putting the source at greater depth and thus generating a large wavelength signal at the surface.

### 4.2 Evaluation of physical variables

We clustered RTM variables by forest type. Coniferous and deciduous forests have distinct distributions and the co-distribution of variables  $Z_{\text{LAI}}$  appears positively correlated with  $Z_{\text{cw}}$ , and negatively

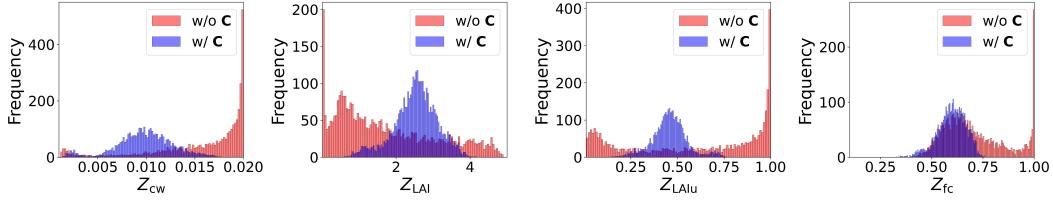


(a) **RTM: Reconstruction accuracy** illustrated for 2 of the spectral bands.

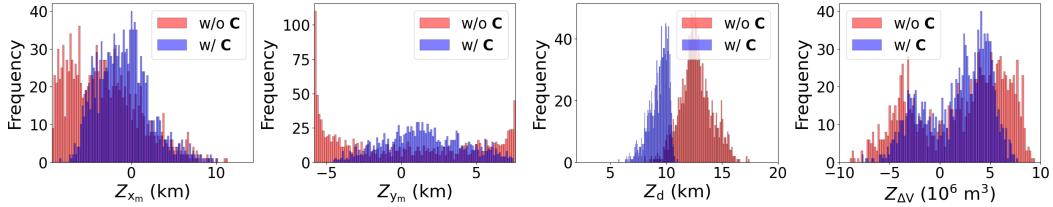


(b) **Mogi: Reconstruction accuracy** of vertical displacements illustrated for 2 of the GNSS stations.

Figure 2: **Reconstruction accuracy without bias correction module displays clear bias.** Blue:  $M_C$ . Red:  $M_B$ . After bias correction, reconstruction for Mogi is not as linear as for RTM, likely due to its simplicity which acts as an information bottleneck in the decoder, highlighting the trade-off between model simplicity and bias correction complexity.

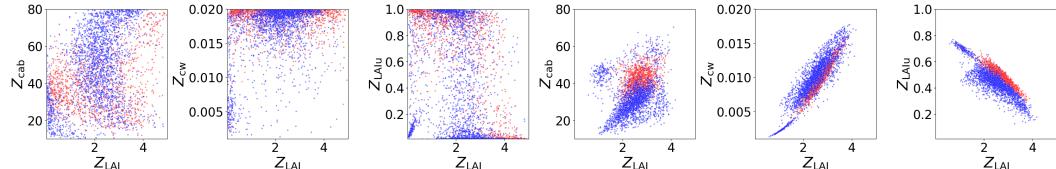


(a) **Distributions of variables:**  $Z_B$ , RTM vs.  $Z_C$ , RTM



(b) **Distributions of variables:**  $Z_B$ , Mogi vs.  $Z_C$ , Mogi

Figure 3: **Bias correction tightened distributions of variables.** Without bias correction, variable distributions (red) are implausible, breaking out from preset bounded ranges. With bias correction (blue) learned distributions are more aligned with the physical sense.

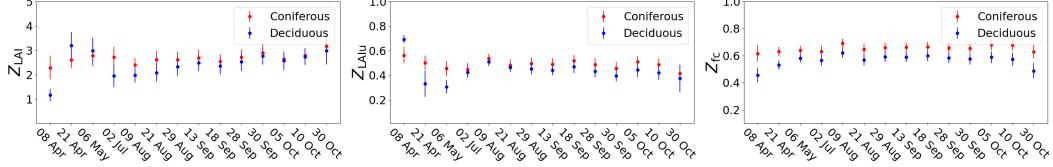


(a) **Co-distributions:**  $Z_{D, \text{RTM}}$

(b) **Co-distributions:**  $Z_{C, \text{RTM}}$

Figure 4: **Pairwise co-distributions** of variables illustrated for  $Z_{\text{LAI}}$ . Red: coniferous forest. Blue: deciduous forest. Our model can learn distinct physical patterns.

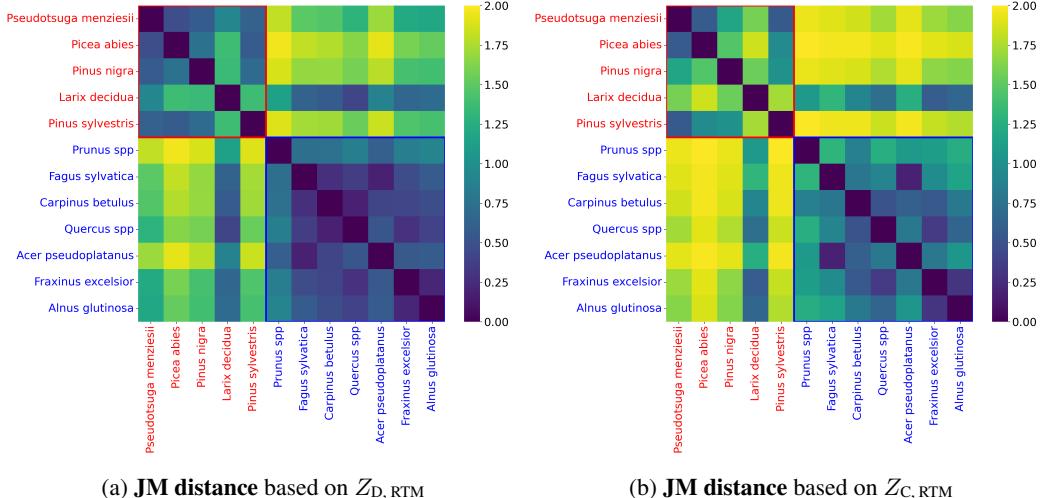
with  $Z_{\text{LAI}_{\text{lu}}}$  under  $\mathbf{M}_{\text{C, RTM}}$  (Fig. 4b). Without bias correction no correlation patterns are visible (Fig. 4a). This makes sense: higher leaf area at the canopy level ( $Z_{\text{LAI}}$ ) indicates plant growth and hence higher water content ( $Z_{\text{cw}}$ ) and chlorophyll content ( $Z_{\text{cab}}$ ) but reduces light penetration, reducing plant growth in the understory, hence lower  $Z_{\text{LAI}_{\text{lu}}}$ . Further illustrations of physical variables are detailed in Appendix A.10.



**Figure 5: Temporal variations of inferred physical parameters  $Z_{\text{C, RTM}}$  illustrated for  $Z_{\text{LAI}}$ ,  $Z_{\text{LAI}_{\text{lu}}}$ , and  $Z_{\text{fc}}$ .  $\mathbf{M}_{\text{C, RTM}}$  effectively captures distinct, temporally smooth, and plausible variations for different forest types, while those from  $\mathbf{M}_{\text{D, RTM}}$  appear to be less smooth (see Appendix A.10.1).**

Temporal variation exhibits more consistent and clearer patterns for some variables (Fig. 5). Understory plant growth ( $Z_{\text{LAI}_{\text{lu}}}$ ) declines from April due to reduced light penetration from increased canopy growth ( $Z_{\text{LAI}}$ ). Coniferous forests exhibit different patterns than deciduous ones, with higher fractional coverage ( $Z_{\text{fc}}$ ) and less variation over time, as would be expected. Temporal variations learned by the  $\mathbf{M}_{\text{D, RTM}}$  appear less consistent (see Appendix A.10.1).

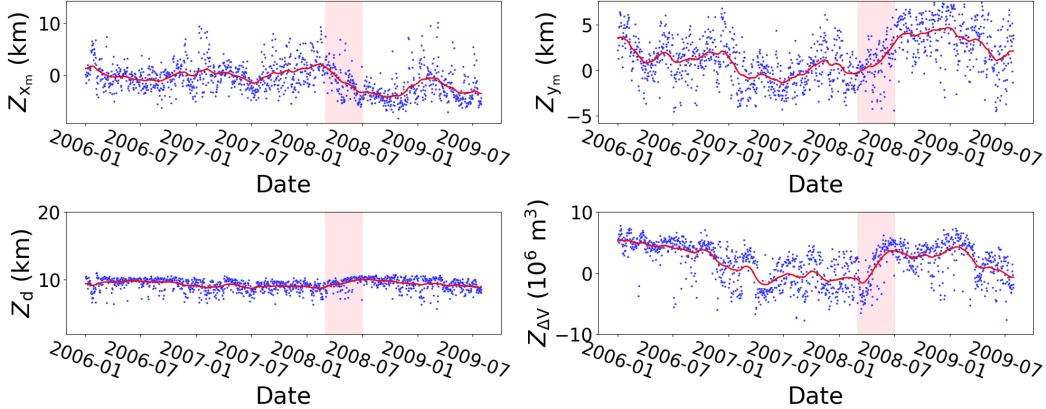
The 7 inferred variables are clustered by species and Jeffereys-Matusita (JM) distance, varying between 0 and 2, is calculated to assess separability of the 12 species. As shown in Fig. 6, our model achieves clear separation between different forest types (deciduous vs. coniferous), and improved separability within forest type, suggesting a degree of disentanglement of inferred variables. Note that the coniferous species ‘*Larix Decidua*’ (European Larch) appears close to the deciduous group, but actually that makes sense as it is one of the few conifers that sheds its needles in the fall.



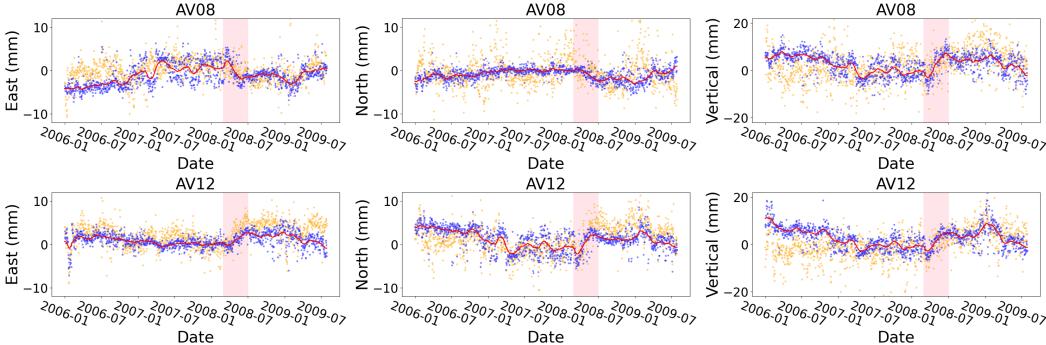
**Figure 6: Pairwise Jeffreys-Matusita (JM) distance between species based on the learned variables. Red: coniferous species. Blue: deciduous species. Species distributions from different forest groups become more disentangled after bias correction.**

For the Mogi model, classical signal processing of the time series and Mogi inversions for 2006–2009 show a transient deformation event due to the inflation of a magma source in early 2008 [Ji and Herring, 2011, Walser et al., 2016]. It lasts about half a year and is characterized by the uplift and radial extension of the surface around the Mogi source. This is consistent with the temporal variation of  $Z_{\Delta V}$  learned by our model. The magnitude of  $Z_{\Delta V}$  and the estimated values of  $Z_d$  (ranging from 6 to 9 km) and the epicenter (located roughly beneath Akutan volcano), which remain relatively constant over time, align with previous estimates Walser et al. [2016] which estimated Mogi parameters using grid search to fit the model to the processed GNSS signals.

Examining two GNSS stations, AV08 and AV12, located on opposite sides of the volcano, we compare their GNSS displacements as reconstructed by  $\mathbf{F}_{\text{Mogi}}$  under the  $\mathbf{M}_{\text{C}, \text{Mogi}}$  model. The model successfully reconstructs the transient signals of the 2008 inflation and the subsequent slow deflation, characterised by specific directional movements, with consistency across stations and directions that align with Ji and Herring [2011] and Walwer et al. [2016]. Note that classical methods [Walwer et al., 2016] required the GNSS signals to be filtered for assumed biases, and transient signals identified by comparing processed signals across GNSS stations, followed by grid search to invert the Mogi model. In contrast, our auto-encoder approach allows  $\mathbf{M}_{\text{C}, \text{RTM}}$  to capture transient signals, estimate Mogi parameters, and correct biases in a single stage without relying on additional data filtering or strong bias assumptions. Furthermore, this inference is achieved through a single forward pass of  $\mathbf{M}_{\text{C}, \text{RTM}}$ , which is trained on datasets with no overlap with the test period.



**Figure 7: Time series of inferred physical parameters.** The red curve is fitted by a Kalman filter to indicate the temporal variations more clearly. The volume change  $Z_{\Delta V}$  successfully captured a major volcanic inflation event in 2008, reaching a peak of approximately  $5 \cdot 10^6 \text{ m}^3$  at a depth of just less than 10 km, consistent with previous findings [Walwer et al., 2016].



**Figure 8: Time series of GNSS displacements** illustrated for two GNSS stations, located at the opposite side of the Mogi source. Orange:  $X_{\text{GNSS}}$ . Blue: Mogi reconstructions using  $Z_{\text{C}, \text{Mogi}}$ . Red: Mogi reconstructions fitted by a Kalman filter. Our model successfully reconstructs the 2008 volcanic inflation transient signals, characterized by radial expansion and vertical uplift, with displacement magnitudes of 10 mm, aligned with prior findings [Ji and Herring, 2011, Walwer et al., 2016].

## 5 Conclusion and Limitations

We have developed an end-to-end approach to learn the inverse of deterministic physical models while correcting for biases. We demonstrated the generalisability of our approach by applying it to physical models from very different domains, each with distinct complexities, differentiability issues, and data dynamics. Extensive analysis shows that our approach has improved reconstruction loss and strongly suggests the plausibility of the recovered physical variables.

Our work suffers from some limitations. First, our approach does not extend to physical models with stochastic outcomes, where the same inputs can lead to different outputs, a much more complex inversion problem. Second, we have only studied two representative physical models. However, we anticipate that our approach can be applied to models in other physical domains. Finally, we use the same bias correction independent of the physical model. While we found this to be effective, given the trade-off between model completeness and complexity necessary for bias correction, future work could use neural architecture search [Ren et al., 2021] to identify the optimal bias correction layer for a specific inverse problem.

**Acknowledgement:** This work was supported by the UKRI Centre for Doctoral Training in Application of Artificial Intelligence to the study of Environmental Risks (reference EP/S022961/1) and Cambridge Centre for Carbon Credits. We would also like to thank Markus Immitzer from Mantle Labs for sharing the Sentinel-2 data with us.

## References

- Andy Adler and David Holder. *Electrical impedance tomography: methods, history and applications*. CRC Press, 2021.
- William A Lahoz and Philipp Schneider. Data assimilation: making sense of earth observation. *Frontiers in Environmental Science*, 2:16, 2014.
- Michael S Zhdanov. *Inverse theory and applications in geophysics*, volume 36. Elsevier, 2015.
- Charles W Groetsch and CW Groetsch. *Inverse problems in the mathematical sciences*, volume 52. Springer, 1993.
- Martin Claverie, Junchang Ju, Jeffrey G Masek, Jennifer L Dungan, Eric F Vermote, Jean-Claude Roger, Sergii V Skakun, and Christopher Justice. The harmonized landsat and sentinel-2 surface reflectance data set. *Remote sensing of environment*, 219:145–161, 2018.
- Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. *Advances in neural information processing systems*, 29, 2016.
- Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations, 2019.
- Abhishek Kumar, Prasanna Sattigeri, and Avinash Balakrishnan. Variational inference of disentangled latent concepts from unlabeled observations, 2018.
- Matthew M Loper and Michael J Black. Opentr: An approximate differentiable renderer. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part VII 13*, pages 154–169. Springer, 2014.
- Hiroharu Kato and Tatsuya Harada. Learning view priors for single-view 3d reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9778–9787, 2019.
- Peter Y Lu, Samuel Kim, and Marin Soljačić. Extracting interpretable physical parameters from spatiotemporal systems using unsupervised learning. *Physical Review X*, 10(3):031056, 2020.
- Yoël Zéraï, Silvia Valero, and Jordi Inglada. Physics-guided interpretable probabilistic representation learning for high resolution image time series. *IEEE Transactions on Geoscience and Remote Sensing*, 2022.
- Gab Abramowitz, Andy Pitman, Hoshin Gupta, Eva Kowalczyk, and Yingping Wang. Systematic bias in land surface models. *Journal of Hydrometeorology*, 8(5):989–1001, 2007.
- J-L Widlowski, B Pinty, M Lopatka, C Atzberger, D Buzica, Michaël Chelle, M Disney, J-P Gastellu-Etchégoity, M Gerboles, N Gobron, et al. The fourth radiation transfer model intercomparison (rami-iv): Proficiency testing of canopy reflectance models with iso-13528. *Journal of Geophysical Research: Atmospheres*, 118(13):6869–6890, 2013.
- Clement Atzberger. Development of an invertible forest reflectance model: The infor-model. In *A decade of trans-european remote sensing cooperation. Proceedings of the 20th EARSeL Symposium Dresden, Germany*, volume 14, pages 39–44, 2000.
- Kiyoo Mogi. Relations between the eruptions of various volcanoes and the deformations of the ground surfaces around them. *Earthq Res Inst*, 36:99–134, 1958.
- Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International conference on learning representations*, 2017.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

Gwynn H Suits. The calculation of the directional reflectance of a vegetative canopy. *Remote Sensing of Environment*, 2:117–125, 1971.

Xiaowen Li and Alan H Strahler. Geometric-optical modeling of a conifer forest canopy. *IEEE Transactions on Geoscience and Remote Sensing*, (5):705–721, 1985.

Paul Segall. Volcano deformation and eruption forecasting. *Geological Society, London, Special Publications*, 380(1):85–106, 2013.

Narendra S Goel. Models of vegetation canopy reflectance and their use in estimation of biophysical parameters from reflectance data. *Remote sensing reviews*, 4(1):1–212, 1988.

B Combal, Frédéric Baret, M Weiss, Alain Trubuil, D Mace, A Pragnere, R Myneni, Y Knyazikhin, and L Wang. Retrieval of canopy biophysical variables from bidirectional reflectance: Using prior information to solve the ill-posed inverse problem. *Remote sensing of environment*, 84(1):1–15, 2003.

P Gong. Inverting a canopy reflectance model using a neural network. *International Journal of Remote Sensing*, 20(1):111–122, 1999.

Martin Schlerf and Clement Atzberger. Inversion of a forest reflectance model to estimate structural canopy variables from hyperspectral remote sensing data. *Remote sensing of environment*, 100(3):281–294, 2006.

S Mostafa Mousavi, Gregory C Beroza, Tapan Mukerji, and Majid Rasht-Behesht. Applications of deep neural networks in exploration seismology: A technical survey. *Geophysics*, 89(1):WA95–WA115, 2024.

Hiroharu Kato, Deniz Beker, Mihai Morariu, Takahiro Ando, Toru Matsuoka, Wadim Kehl, and Adrien Gaidon. Differentiable rendering: A survey. *arXiv preprint arXiv:2006.12057*, 2020.

Georgios Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. Learning to estimate 3d human pose and shape from a single color image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 459–468, 2018.

Yoël Zérah, Silvia Valero, and Jordi Ingla. Physics-constrained deep learning for biophysical parameter retrieval from sentinel-2 images: inversion of the prosail model. *Available at SSRN 4671923*, 2023.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Tommaso Jucker, John Caspersen, Jérôme Chave, Cécile Antin, Nicolas Barbier, Frans Bongers, Michele Dalponte, Karin Y van Ewijk, David I Forrester, Matthias Haeni, et al. Allometric equations for integrating remote sensing imagery into forest monitoring programmes. *Global change biology*, 23(1):177–190, 2017.

Damian Walwer, Eric Calais, and Michael Ghil. Data-adaptive detection of transient deformation in geodetic networks. *Journal of Geophysical Research: Solid Earth*, 121(3):2129–2152, 2016.

OpenAI. Gpt-4 technical report, 2023.

Matthias Drusch, Umberto Del Bello, Sébastien Carlier, Olivier Colin, Veronica Fernandez, Ferran Gascon, Bianca Hoersch, Claudia Isola, Paolo Laberinti, Philippe Martimort, et al. Sentinel-2: Esa’s optical high-resolution mission for gmes operational services. *Remote sensing of Environment*, 120:25–36, 2012.

Geoffrey Blewitt, William Hammond, et al. Harnessing the gps data explosion for interdisciplinary science. *Eos*, 99, 2018.

Kang Hyeun Ji and Thomas A Herring. Transient signal detection using gps measurements: Transient inflation at akutan volcano, alaska, during early 2008. *Geophysical Research Letters*, 38(6), 2011.

Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Xiaojiang Chen, and Xin Wang. A comprehensive survey of neural architecture search: Challenges and solutions. *ACM Computing Surveys (CSUR)*, 54(4):1–34, 2021.

Adriano Gualandi, Enrico Serpelloni, and Maria Elina Belardinelli. Blind source separation problem in gps time series. *Journal of Geodesy*, 90(4):323–341, 2016.

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Related Work</b>	<b>2</b>
2.1	Disentangled representation learning . . . . .	2
2.2	Inverse problems in physical sciences . . . . .	3
2.3	Autoencoder as an approach to inversion . . . . .	3
<b>3</b>	<b>Methods</b>	<b>3</b>
3.1	Learning the inverse of physical models . . . . .	3
3.2	Physical models for inversion . . . . .	4
3.3	End-to-end learning with physical models . . . . .	5
3.4	Datasets . . . . .	5
3.5	Baseline . . . . .	5
<b>4</b>	<b>Results</b>	<b>6</b>
4.1	Effects of bias correction . . . . .	6
4.2	Evaluation of physical variables . . . . .	6
<b>5</b>	<b>Conclusion and Limitations</b>	<b>9</b>
<b>A</b>	<b>Appendix</b>	<b>15</b>
A.1	Making a fully differentiable physical model for deep learning . . . . .	15
A.1.1	Conversion of $\mathbf{F}_{\text{RTM}}$ to PyTorch assisted by GPT-4 . . . . .	15
A.1.2	Validating the PyTorch implementation of $\mathbf{F}_{\text{RTM}}$ . . . . .	16
A.2	Bypassing instability points in the training loop . . . . .	16
A.2.1	Numerical instability of $\mathbf{F}_{\text{RTM}}$ . . . . .	16
A.2.2	Forward pass . . . . .	16
A.2.3	Backpropagation . . . . .	17
A.2.4	Stabilized training process . . . . .	17
A.3	Biophysical variables of $\mathbf{F}_{\text{RTM}}$ . . . . .	18
A.4	Sentinel-2 dataset: spectral bands, temporal, and species information . . . . .	18
A.5	Mathematical formulations of $\mathbf{F}_{\text{Mogi}}$ . . . . .	19
A.6	GNSS dataset . . . . .	20
A.6.1	GNSS stations around the Akutan volcano . . . . .	20
A.6.2	Assumed components of GNSS time series . . . . .	20
A.7	Smoothness term added to MSE loss for Mogi inversion . . . . .	21
A.8	Implementation details . . . . .	21
A.9	Extended results for bias correction . . . . .	22
A.9.1	Bias correction for $\mathbf{F}_{\text{RTM}}$ . . . . .	22
A.9.2	Bias correction for $\mathbf{F}_{\text{Mogi}}$ . . . . .	24

A.10 Extended results for the evaluation of physical variables . . . . .	28
A.10.1 Retrieved biophysical variables of $\mathbf{F}_{\text{RTM}}$ . . . . .	28
A.10.2 Reconstructed GNSS signals based on $Z_{C,\text{Mogi}}$ . . . . .	32

## A Appendix

### A.1 Making a fully differentiable physical model for deep learning

#### A.1.1 Conversion of $\mathbf{F}_{\text{RTM}}$ to PyTorch assisted by GPT-4

The original modules of  $\mathbf{F}_{\text{RTM}}$  are extracted from a vegetation app called EnMap. It is implemented using Numpy arrays and operations. To track the computational graph and enable gradient back-propagation, it was reimplemented using PyTorch operations. However,  $\mathbf{F}_{\text{RTM}}$  is not inherently differentiable, and as a complicated physical model, converting it in PyTorch manually was challenging. In light of the recent development of large language models (LLMs), we decided to utilize GPT-4 [OpenAI, 2023] to assist in the conversion from NumPy to PyTorch. While GPT-4 reduced a significant amount of repetitive work, we ran into the following limitations:

GPT-4 is useful for converting common operations from NumPy into PyTorch. We used it to successfully convert 1,742 lines of code across various scripts from the original implementation to PyTorch (e.g. Listing 1 and Listing 2). However, some operations e.g. the exponential integral function from SciPy, have no equivalent in PyTorch (Listing 2). To calculate their derivative, we implemented our own backward functions to define derivatives.

Additionally, GPT-4 only accepts text input of limited length, which means it can only convert a script snippet piece by piece. As the conversation lengthens, it made mistakes. For instance, it converted `numpy.radians` to `torch.radians`, which does not exist in PyTorch. Thus, it was still important to carefully check the conversion, run unit tests, and work on compiling all the scripts together. Interestingly, GPT allows for prompt engineering, which means we can provide feedback to refine its conversions, such as requesting it to highlight uncertain parts for review in subsequent conversions.

```

1 n = PD_refractive
2 k = (np.outer(Cab, PD_k_Cab) + np.outer(Car, PD_k_Car) + np.outer(Anth,
   , PD_k_Anth) +
   np.outer(Cbrown, PD_k_Brown) + np.outer(Cw, PD_k_Cw) + np.outer(
   Cm, PD_k_Cm)) / N[:, np.newaxis]
3
4 ind_k0_row, ind_k0_col = np.where(k == 0)
5
6 if len(ind_k0_row) > 0:
7     k[ind_k0_row, ind_k0_col] = np.finfo(float).eps
9 trans = (1 - k) * np.exp(-k) + (k ** 2) * exp1(k)

```

Listing 1: **Example code of  $\mathbf{F}_{\text{RTM}}$  initially implemented using NumPy.** It features an exponential integral function `exp1` in line 9. Additionally, it's important to highlight that in line 3, the structure parameter `N` functions as a denominator in the calculation, therefore, it cannot be zero.

```

1 n = PD_refractive
2 k = (torch.outer(Cab, PD_k_Cab) + torch.outer(Car, PD_k_Car) + torch.
   outer(Anth, PD_k_Anth) +
   torch.outer(Cbrown, PD_k_Brown) + torch.outer(Cw, PD_k_Cw) +
   torch.outer(Cm, PD_k_Cm)) / N.unsqueeze(-1)
3
4 ind_k0_row, ind_k0_col = torch.where(k == 0)
5
6 if len(ind_k0_row) > 0:
7     k[ind_k0_row, ind_k0_col] = torch.finfo(float).eps
8

```

```

9 trans = (1 - k) * torch.exp(-k) + (k ** 2) * torch.from_numpy(exp1(k.
    numpy()))

```

**Listing 2: Conversion of  $\mathbf{F}_{\text{RTM}}$  to PyTorch by GPT.** It is followed by feedback on `exp1` to raise user's attention. Here is an excerpt: "Please note that the `exp1` function is not available in PyTorch as of my last training cut-off in September 2021. We can import it from the `scipy.special` library, but we have to convert the tensor to `numpy` before using it and convert the result back to a tensor. It's not ideal in terms of performance, but I'm afraid there's no other option for now. Please check if there is an updated version of PyTorch that includes this function."

This experiment was done in May 2023, when GPT was still in its early phase of development. It has had several features updated since then. For instance, it can now accept longer text inputs. Nonetheless, we believe the limitations discussed above still largely hold true and are worth considering when converting legacy implementations of other complex physical models.

### A.1.2 Validating the PyTorch implementation of $\mathbf{F}_{\text{RTM}}$

To validate the implementation of  $\mathbf{F}_{\text{RTM}}$  in PyTorch against its original version in NumPy, we randomly sampled 10,000 sets of variables, passed them through each implementation, and compared their outputs (Tab. 5). The mismatch rate of the outputs is 0.457% when the absolute tolerance is set to 1e-5. The maximum absolute difference over all 130000 output reflectances from the two implementations is 3.050e-5 wherein the physical unit of reflectance here is 1. This indicates that  $\mathbf{F}_{\text{RTM}}$  implemented in PyTorch is substantially equivalent to its original version and, therefore can be used in subsequent tasks. The PyTorch implementation for the RTM has been released on GitHub (<https://github.com/yihshe/MAGIC.git>) as a public Python library to facilitate its integration into other deep-learning workflows.

**Table 5: Unit test of the outputs of our PyTorch implementation of  $\mathbf{F}_{\text{RTM}}$  against the original NumPy version.** The total number of elements to compare is 130000 as we simulate the same set of 10000 samples for both implementations, each with 13 spectral bands of Sentinel-2.

Total Elements	Mismatched Elements	Absolute Tolerance	Mismatch Ratio	Max Absolute Difference
130000	594	1e-5	0.457%	3.050e-5

## A.2 Bypassing instability points in the training loop

### A.2.1 Numerical instability of $\mathbf{F}_{\text{RTM}}$

With  $\mathbf{F}_{\text{RTM}}$  now implemented in PyTorch, we can backpropagate gradients through this physical model. However, the model is sensitive to input variable ranges and is non-differentiable at certain points. Furthermore, complex operations like exponential, logarithm, and square root, often lead to numerical instability during derivative computation, even though these operations are theoretically differentiable. To solve these issues, we have developed the following methods to stabilize training when using a complex physical model like  $\mathbf{F}_{\text{RTM}}$ .

### A.2.2 Forward pass

By definition, input variables  $Z_{\text{RTM}}$  are non-negative values with physical meanings (Tab. 1). In practice,  $\mathbf{F}_{\text{RTM}}$  can fail in the forward pass when  $Z_{\text{RTM}}$  from other value ranges are provided (e.g. structure parameter  $Z_N$  in Listing 1). The same concept applies to  $\mathbf{F}_{\text{Mogi}}$ . Thus, for both models, we choose to initially infer normalized variables  $\Lambda \in [0, 1]$  and subsequently map these to variables  $Z \in [Z_{\min}, Z_{\max}]$  in their original scale, according to their value ranges (Tabs. 1 and 2).

$$Z = (Z_{\max} - Z_{\min}) \cdot \Lambda + Z_{\min} \quad (6)$$

This approach not only facilitates a successful forward pass of physical models but also narrows down the search space for optimization.

### A.2.3 Backpropagation

Despite encoding the forward pass successfully, backpropagation remained unstable for  $\mathbf{F}_{\text{RTM}}$ . For successful learning, it is necessary to backpropagate gradients through the physical model. The aforementioned caveats of  $\mathbf{F}_{\text{RTM}}$  resulted in backpropagation failures, leading to NaN gradient and then NaN loss. During experiments to stabilize the backpropagation, we noticed that NaN gradients, when they initially appear, are always in association with specific sets of  $\mathbf{F}_{\text{RTM}}$  variables. Ideally, one would apply constraints to specific  $\mathbf{F}_{\text{RTM}}$  operations causing these NaN gradients. However, working out such constraints seemed challenging unless we understood the semantics of  $\mathbf{F}_{\text{RTM}}$  and its differentiability thoroughly, a complex task.

We, therefore, introduced a simple workaround: since it is the NaN gradients that cause subsequent learning failure, we replace them with a small random value, sampled from a uniform [0,1] distribution and scaled by 1e-7, whenever these gradients first appear. The specific workflow of the algorithm is described in algorithm 1

---

#### Algorithm 1 Gradient stabilization

---

```

Calculate gradients
Initialize list grads, containing gradients of all model parameters where gradients exist and contain
any NaN values
if grads is not empty then
    Set a small constant epsilon equal to 1e-7
    for each v in grads do
        Generate random values of the same shape as v, scaled by epsilon
        Create a mask where v is NaN or equals 0
        Replace values in v where the mask is True with corresponding random values
    end for
end if
Update gradients

```

---

### A.2.4 Stabilized training process

Our methods effectively stabilized the learning of  $\mathbf{M}_{B,\text{RTM}}$  and  $\mathbf{M}_{C,\text{RTM}}$  in the autoencoder framework, despite instability of  $\mathbf{F}_{\text{RTM}}$  during backpropagation. As can be seen in Fig. 9, both training and validation losses of  $\mathbf{M}_{C,\text{RTM}}$  converge over the epochs. Therefore, by preventing the propagation of NaN gradients to earlier layers and allowing the normal progression of the forward pass, the stabilizer has effectively enabled the training process to bypass instability points in the optimization space and allowed a continuous search for optimal points.

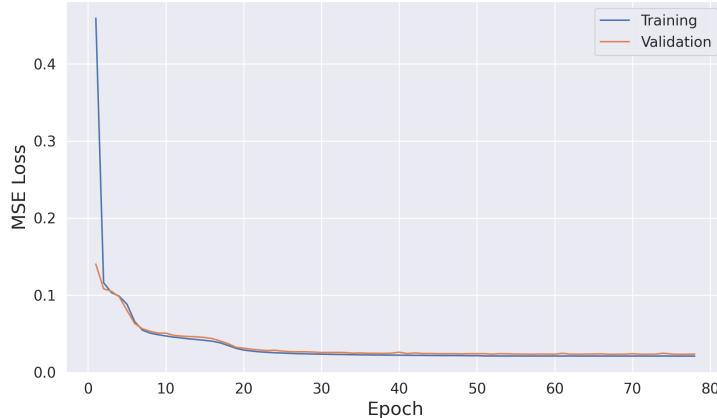


Figure 9: **Stabilized training of  $\mathbf{M}_{C,\text{RTM}}$ .** Our algorithm to update the gradients has overcome the instability of  $\mathbf{F}_{\text{RTM}}$  during backpropagation and allowed the convergence of training loss.

### A.3 Biophysical variables of $\mathbf{F}_{\text{RTM}}$

The input of  $\mathbf{F}_{\text{RTM}}$  consists of biophysical variables of three hierarchical levels. Note that the original  $\mathbf{F}_{\text{RTM}}$  does not have fractional coverage  $Z_{\text{fc}}$  as an input variable. We include  $Z_{\text{fc}}$  as one of the seven variables to be learned directly by our model, which will be used to infer crown diameter  $Z_{\text{cd}}$  and height  $Z_{\text{h}}$  based on derived equations. As the fractional coverage is jointly defined by the stem density and crown diameter within each unit hectare (or  $10,000 \text{ m}^2$ ),  $Z_{\text{cd}}$  can be derived given  $Z_{\text{fc}}$  and  $Z_{\text{cd}}$  using eq. (7). Furthermore, to derive  $Z_{\text{h}}$ , we fit an allometric equation (Fig. 10, eq. (8)) using the samples of temperate forests from the global allometric database [Jucker et al., 2017].  $R^2$  of the derived equation between  $Z_{\text{h}}$  and  $Z_{\text{cd}}$  is 0.383 (eq. (8)).

Table 6: Overview of the biophysical variables of  $\mathbf{F}_{\text{RTM}}$ . These variables can be attributed to three hierarchical levels. 7 variables are learned directly. \*  $Z_{\text{cd}}$  and  $Z_{\text{h}}$  will be inferred from  $Z_{\text{fc}}$  using eq. (7) and eq. (8), respectively.

Group	Variable	Acronym	To Learn	Default Value	Sample Range	
					Min	Max
Background	Soil brightness factor	psoil	✗	0.8	-	-
	Structure Parameter	N	✓	-	1	3
	Chlorophyll A+B	cab	✓	-	10	80
	Water Content	cw	✓	-	0.001	0.02
	Dry Matter	cm	✓	-	0.005	0.05
	Carotenoids	car	✗	10	-	-
	Brown Pigments	cbrown	✗	0.25	-	-
	Anthocyanins	anth	✗	2	-	-
	Proteins	cp	✗	0.0015	-	-
	Cabon-based Constituents	cbc	✗	0.01	-	-
Leaf Model	Leaf Area Index	LAI	✓	-	0.01	5
	Leaf Angle Distribution	typeLIDF	✗	Beta Distribution	-	-
	Hot Spot Size	hspot	✗	0.01	-	-
	Observation Zenith Angle	tto	✗	0	-	-
	Sun Zenith Angle	tts	✗	30	-	-
	Relative Azimuth Angle	psi	✗	0	-	-
Canopy Model	Undergrowth LAI	LAIu	✓	-	0.01	1
	Stem Density	sd	✗	500	-	-
	Fractional Coverage	fc	✓	-	0.1	1
	Tree Height	h	✓	*	*	*
	Crown Diameter	cd	✓	*	*	*
Forest Model						

$$Z_{\text{cd}} = 2 \cdot \sqrt{\frac{Z_{\text{fc}} \cdot 10000}{\pi \cdot Z_{\text{sd}}}} \quad (7)$$

$$Z_{\text{h}} = \exp(2.117 + 0.507 \cdot \ln(Z_{\text{cd}})) \quad (8)$$

### A.4 Sentinel-2 dataset: spectral bands, temporal, and species information

The detailed information of the spectral bands of  $X_{\text{S2}}$  can be viewed in Tab. 7. Statistics of the  $X_{\text{S2}}$  can be viewed in Tab. 8. These spectra were sampled from individual sites covering a time sequence of 14 timestamps. Note that our current approach to inverting  $\mathbf{F}_{\text{RTM}}$  does not integrate temporal information into the learning process, although temporal variations are evaluated. Such temporal information could serve as useful prior knowledge to boost the model's performance, especially to ensure the consistency of temporal variations, which we will consider in future works.

All samples from  $X_{\text{S2}}$  cover both coniferous and deciduous forests consisting of 5 and 7 species, respectively:

- **Coniferous forest:** Pseudotsuga Menziesii, Picea Abies, Pinus Nigra, Larix Decidua, Pinus Sylvestris.

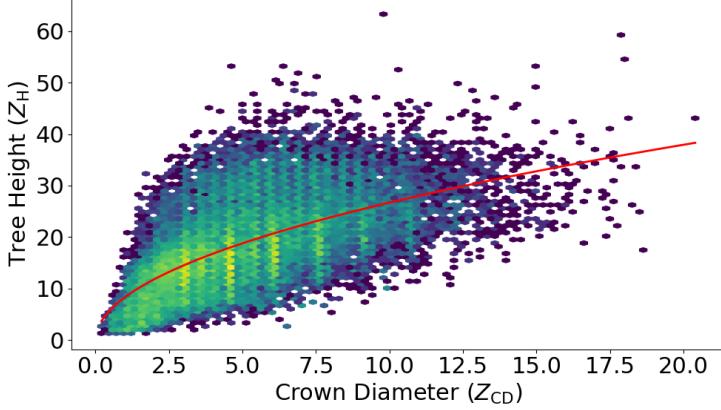


Figure 10: **Allometric relationship between canopy height  $Z_h$  and crown diameter  $Z_{cd}$ .** We fit an allometric equation between  $Z_h$  and  $Z_{cd}$  (eq. (8)).  $R^2$  of the fitted equation is 0.383.

Table 7: **Sentinel-2 bands** to use. *VNIR* stands for *Visible and Near Infrared*. *SWIR* stands for *Short Wave Infrared*.

Band	B2	B3	B4	B5	B6	B7	B8	B8a	B9	B11	B12
Resolution	10 m	10 m	10 m	20 m	20 m	20 m	10 m	20 m	60 m	20 m	20 m
Central Wavelength	490 nm	560 nm	665 nm	705 nm	740 nm	783 nm	842 nm	865 nm	940 nm	1610 nm	2190 nm
Description	Blue	Green	Red	VNIR	VNIR	VNIR	VNIR	VNIR	SWIR	SWIR	SWIR

- **Deciduous forest:** Prunus Spp, Fagus Sylvatica, Carpinus Betulus, Quercus Spp, Acer Pseudoplatanus, Fraxinus Excelsior, Alnus Glutinosa.

### A.5 Mathematical formulations of $\mathbf{F}_{\text{Mogi}}$

$\mathbf{F}_{\text{Mogi}}$  describes the displacement field on the surface resulting from a spherical pressure source, typically a magma chamber, at depth. The inverse problem of  $\mathbf{F}_{\text{Mogi}}$  deals with estimating the location and volume change of magma chamber activities given the observed displacements at ground stations. In this project,  $\mathbf{F}_{\text{Mogi}}$  to use is further simplified by assuming only a point pressure source. Specifically, for  $N$  ground stations located at  $\{x_i, y_i, z_i, i = 1, \dots, N\}$ , the displacement field  $X_{\text{GNSS}} = \{x_{e,i}, x_{n,i}, x_{v,i}, i = 1, \dots, N\}$  at the surface along the east, north, and vertical directions due to the subsurface volumetric change of a Mogi source is estimated by:

$$\hat{x}_{e,i} = \frac{\alpha(x_i - Z_{x_m})}{R_i^3}, \quad (9)$$

$$\hat{x}_{n,i} = \frac{\alpha(y_i - Z_{y_m})}{R_i^3}, \quad (10)$$

$$\hat{x}_{v,i} = \frac{\alpha(z_i - Z_d)}{R_i^3} \quad (11)$$

where  $Z_{x_m}$  and  $Z_{y_m}$  are the horizontal coordinates of the Mogi source, and  $Z_d$  is its depth.  $\alpha$  is the scaling factor that includes the volume change  $Z_{\Delta V}$  and the material properties (Poisson's ratio of the medium), given by:

$$\alpha = \frac{(1 - \nu)Z_{\Delta V}}{\pi} \quad (12)$$

Table 8: **Information of the Sentinel-2 dataset  $X_{S2}$**

Total Number of Spectra	Number of Individual Sites	Number of Dates	Number of Species
17962	1283	14	12

and  $R_i$  is the radial distance from the Mogi source point to the station at  $(x_i, y_i, 0)$  (the equations are valid for a half-space, so elevation is neglected):

$$R_i = \sqrt{(x_i - x_m)^2 + (y_i - y_m)^2 + Z_d^2} \quad (13)$$

For the inversion problem, we assume the following parameters are known:

- $\nu$ , Poisson's ratio, usually set at 0.25, a typical value for the Earth's crust.
- $\{x_i, y_i | i = 1, \dots, N\}$ , locations of the GNSS stations.

Thus, our goal is to estimate the location  $(Z_{x_m}, Z_{y_m}, Z_d)$  and volume change  $Z_{\Delta V}$  of the Mogi source, given the measured displacements recorded by the GNSS stations  $X_{\text{GNSS}} = \{u_{e,i}, u_{n,i}, u_{v,i} | i = 1, \dots, N\}$ .

## A.6 GNSS dataset

### A.6.1 GNSS stations around the Akutan volcano

Surface displacement is monitored via continuous GNSS (Global Navigation Satellite System) stations that produce daily position time series (<http://geodesy.unr.edu>, Blewitt et al. [2018]). For the Akutan volcano, we have 12 GNSS stations within the region (Fig. 11). From these stations, we have compiled daily time-series GNSS displacement data spanning from 2006 to 2024, which we will use to test our inversion approach. It is common for GNSS stations to occasionally miss data due to various issues such as power outages or instrument replacements. Approximately 20% of our raw time series data is missing, which, although significant, is still considered good in terms of GNSS observations. We have employed a variational Bayesian Independent Component Analysis [Gualandi et al., 2016] to fill up these data gaps using existing observations to make sure we have a complete daily time series.

Our model takes the displacement field observed at each single day as a training sample. However, we can check the plausibility of the inversion by evaluating the temporal variations of the retrieved variables and reconstructed signals comparing them against existing literature [Ji and Herring, 2011, Walwer et al., 2016].

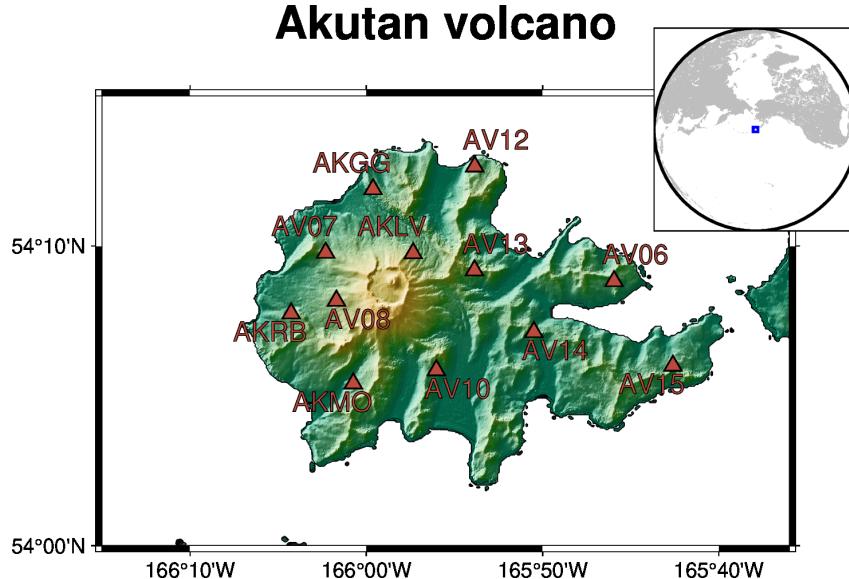


Figure 11: Locations of the 12 GNSS stations around the Akutan Volcano.

### A.6.2 Assumed components of GNSS time series

In addition to deformations due to a Mogi source, the surface displacement observed by a GNSS station can be influenced by other physical processes and noise sources. In simplified form, a

displacement  $u(t)$  of a station along certain direction at time  $t$  can be modeled as [Walwer et al., 2016]:

$$u(t) = S_{\text{linear}}(t) + S_{\text{annual}}(t) + S_{\text{semiannual}}(t) + S_{\text{eruption}}(t) + \epsilon_{\text{pink}} + \epsilon_{\text{white}} \quad (14)$$

Where:

$$S_{\text{linear}}(t) = q + m \cdot t \quad (15)$$

$$S_{\text{annual}}(t) = A_1 \cdot \sin\left(\frac{2\pi t}{T_1}\right) + B_1 \cdot \cos\left(\frac{2\pi t}{T_1}\right) \quad (16)$$

$$S_{\text{semiannual}}(t) = A_2 \cdot \sin\left(\frac{2\pi t}{T_2}\right) + B_2 \cdot \cos\left(\frac{2\pi t}{T_2}\right) \quad (17)$$

$$S_{\text{eruption}}(t) = C \cdot Z_{\Delta V,t} \quad (18)$$

$S_{\text{linear}}$  represents the linear trend, often due to tectonic plate movement;  $S_{\text{annual}}$  and  $S_{\text{semiannual}}$  represent the annual and semiannual seasonality of the signal, potentially due to hydrological processes; and  $S_{\text{eruption}}$  accounts for the deformation due to Mogi source since the eruption event happening at  $t_s$  and depends on the volume change  $Z_{\Delta V,t}$  at time step  $t$ .  $\epsilon_{\text{pink}}$  and  $\epsilon_{\text{white}}$  are pink and white noises, respectively, commonly observed in natural phenomena.

The volcanic deformation  $S_{\text{eruption}}(t)$  can be small (usually at mm level) if compared to other sources of deformation in eq. (14) (mm to cm level), making the bias correction both an essential ingredient to get accurate results and a challenging task due to the incompleteness of  $\mathbf{F}_{\text{Mogi}}$ .

### A.7 Smoothness term added to MSE loss for Mogi inversion

Compared to  $X_{S2}$ ,  $X_{\text{GNSS}}$  exhibits stronger temporal dynamics as a daily time series. While our model takes the displacement field of each day as input, enhancing the smoothness of the Mogi reconstructions along the temporal dimension will align the results more closely with the expectation that the deformation from the volcanic source ( $S_{\text{eruption}}$ ) should demonstrate temporal continuity, rather than being influenced by the patterns of systematic biases from other components in eq. (14). To enhance the temporal continuity of the Mogi reconstructions, we structure the data such that each batch contains mini-sequences sampled from  $T$  consecutive days. We have also introduced an additional smoothness term to  $\mathbf{M}_{C,\text{Mogi}}$ , which is expressed as follows:

$$\mathcal{L}_{\text{total}}(\theta_{\mathbf{E}_C}, \theta_C) = \mathcal{L}_{\text{MSE}}(\theta_{\mathbf{E}_C}, \theta_C) + \lambda \sum_{j=1}^B \sum_{t=1}^{T-1} \|x_{\mathbf{F},C,j}^{(t+1)} - x_{\mathbf{F},C,j}^{(t)}\|^2 \quad (19)$$

where the sequence size  $T$  is the number of time steps in each sequence, and the batch size  $B$  is the number of sequences in each batch. In our case,  $T = 5$  and  $B = 12$ , leading to a total of 60 samples per batch, matching the training setup used for the classical auto-encoder  $\mathbf{M}_{A,\text{Mogi}}$ . The parameter  $\lambda$  is set at 0.01 to ensure minimal dramatic changes within the 5-day window, reducing noise and ensuring continuity in the Mogi reconstruction (Fig. 12). The same smoothness term is applied during the training process of  $\mathbf{M}_{B,\text{Mogi}}$ .

### A.8 Implementation details

Both  $X_{S2}$  and  $X_{\text{GNSS}}$  are further split into the train, validation, and test sets. For  $X_{S2}$ , samples from the same individual site are placed in the same set to avoid data leakage. For  $X_{\text{GNSS}}$ , we have saved the GNSS time series from 2006 to 2009 as a test set to cover the 2008 event for validation. The batch size is set to 64 for loading  $X_{S2}$  and 60 for loading  $X_{\text{GNSS}}$ . Training loss is MSE and for  $\mathbf{M}_{B,\text{Mogi}}$  and  $\mathbf{M}_{C,\text{Mogi}}$  with an added regularizer to smooth the reconstructed time series from  $\mathbf{M}_{C,\text{Mogi}}$  (eq. (19)). We used the Adam optimization, setting the initial learning rate at 0.001 and weight decay at 0.0001. The maximum number of epochs is 100 and the learning rate is reduced by a factor of 10 after 50 epochs; training stops if  $\mathcal{L}_{\text{MSE}}$  on the validation set does not improve after an additional 10 epochs. All experiments were conducted on a system equipped with an AMD EPYC 7702 64-Core Processor and 1TB of RAM. Training times varied depending on the model complexity. Specifically, models that require backpropagation through  $\mathbf{F}_{\text{RTM}}$  (such as  $\mathbf{M}_{B,\text{RTM}}$  and  $\mathbf{M}_{C,\text{RTM}}$ ) took approximately 3 hours to train. In contrast, training times for all other models did not exceed half an hour.

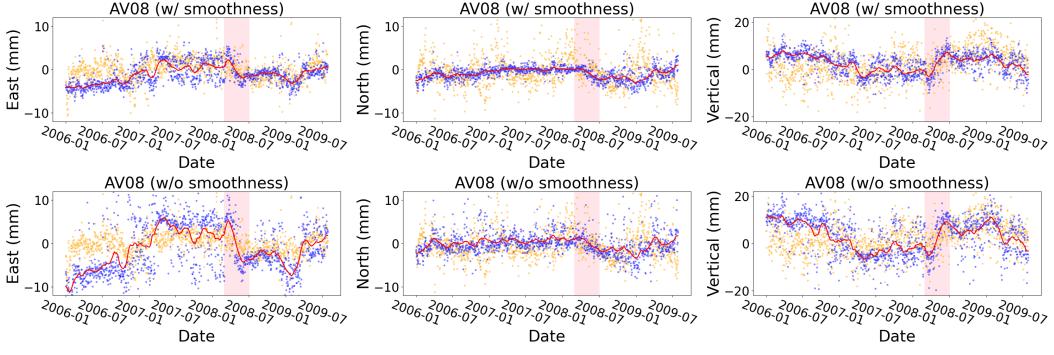


Figure 12: **The smoothness term has enhanced the temporal continuity of Mogi reconstruction** illustrated for AV08. Orange:  $X_{\text{GNSS}}$ . Blue: Mogi reconstructions using  $Z_{C,\text{Mogi}}$ .

## A.9 Extended results for bias correction

### A.9.1 Bias correction for $F_{\text{RTM}}$

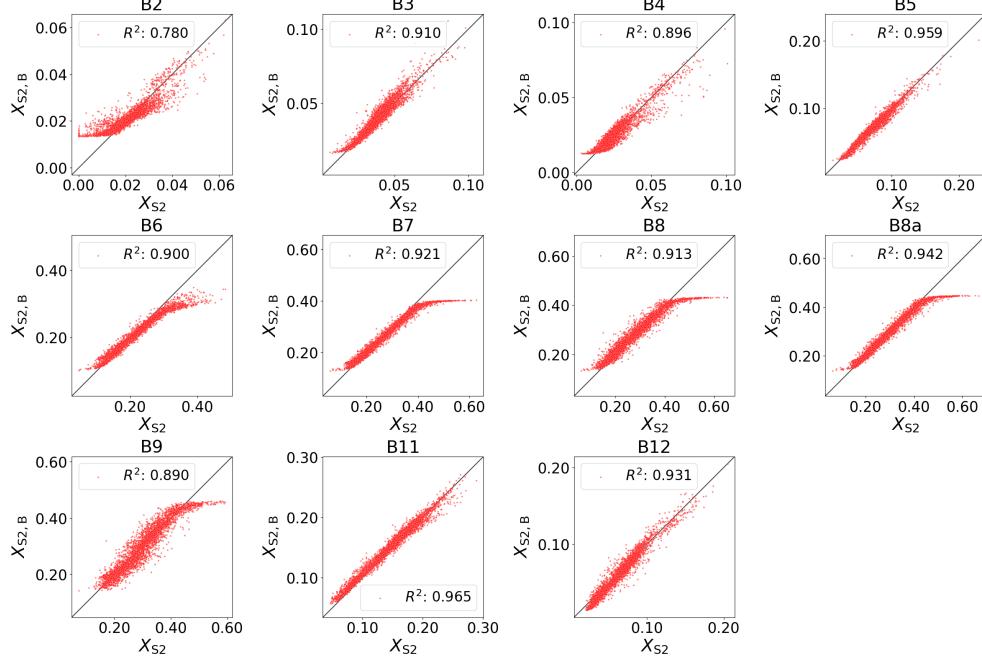


Figure 13: **Reconstruction accuracy of  $X_{B,\text{RTM}}$  displays clear bias without bias correction module.**

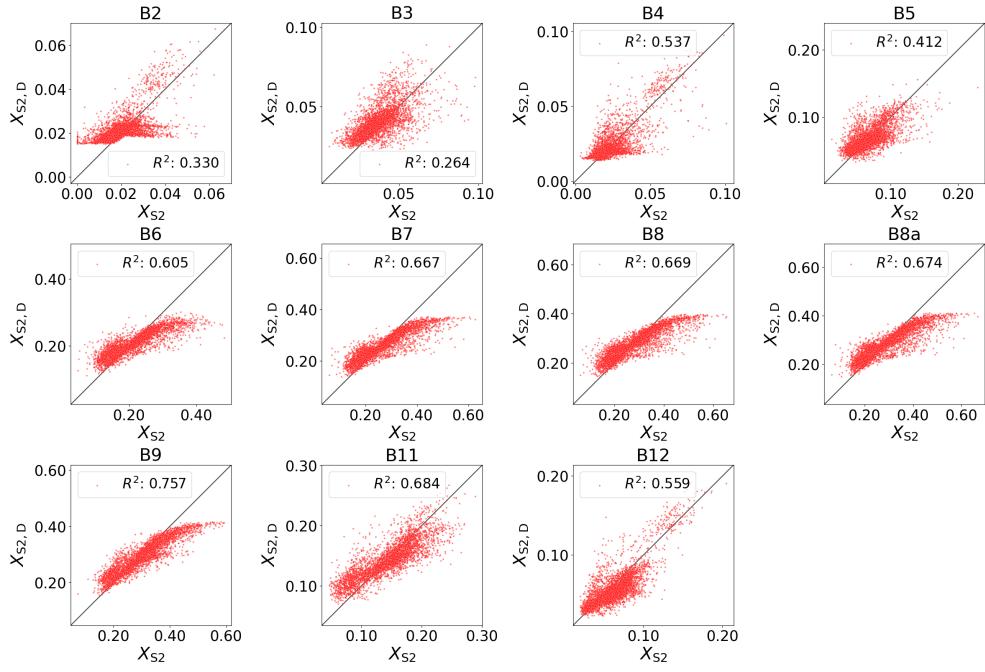


Figure 14: **Reconstruction accuracy of  $X_{D,RTM}$  displays clear bias without bias correction module**, showing a similar pattern of bias as Fig. 14.

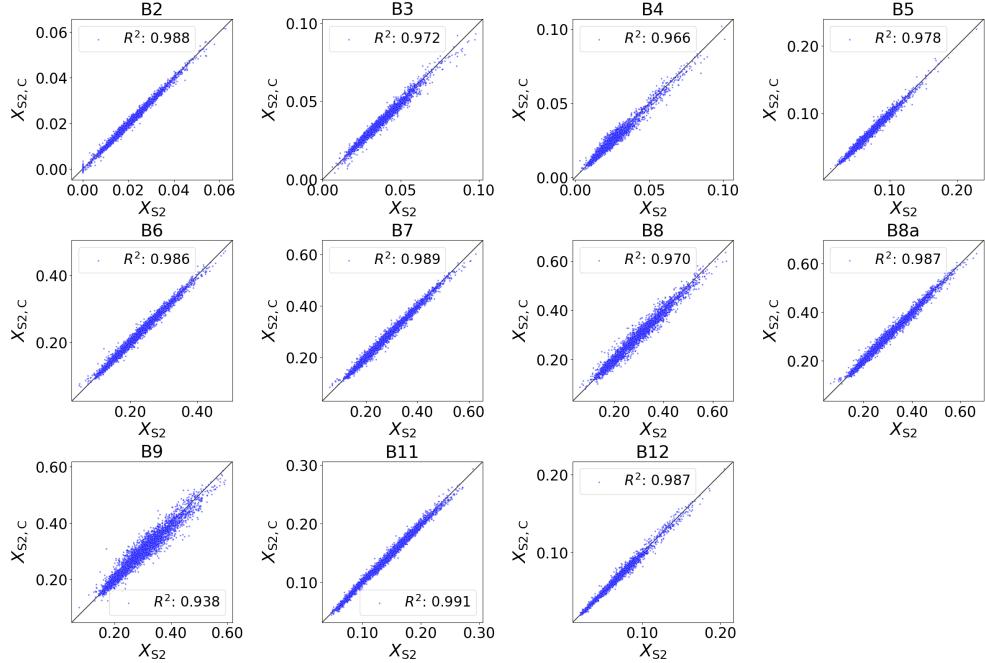


Figure 15: **Reconstruction accuracy of  $X_{C,RTM}$  significantly improved after bias correction.**

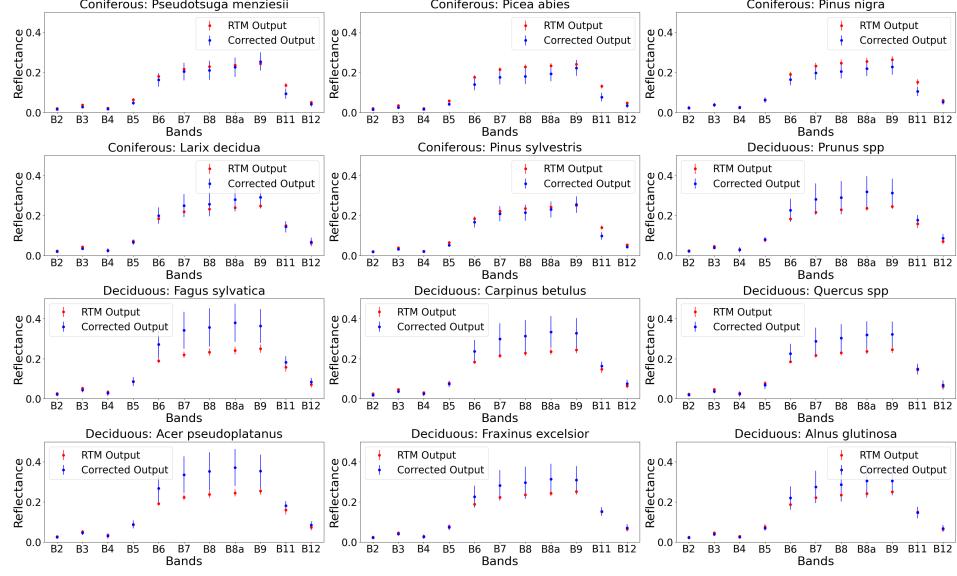


Figure 16: **Reconstructed spectral signatures based on  $Z_{C,RTM}$  and clustered by species**, which display distinct patterns between forest types. Notably, the bias correction for ‘*Larix decidua*’ displays a pattern similar to those of deciduous forest, which is consistent with our findings in Fig. 6b.

#### A.9.2 Bias correction for $\mathbf{F}_{Mogi}$

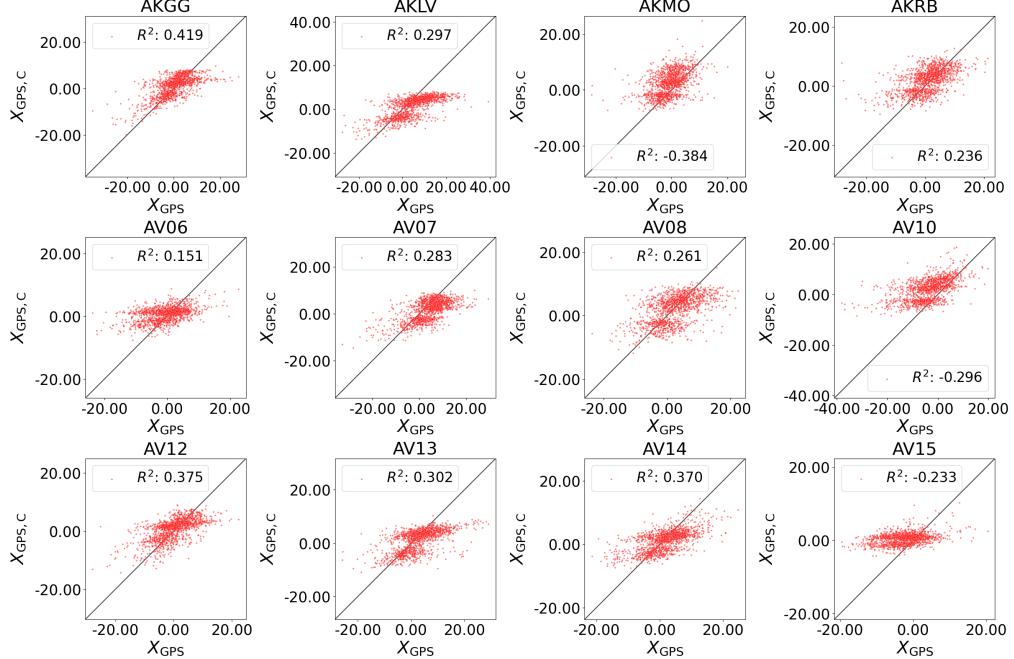


Figure 17: **Reconstruction accuracy of  $X_{B,Mogi}$  illustrated for vertical displacements, which displays clear bias and is more complex than those shown by  $\mathbf{F}_{RTM}$ .**

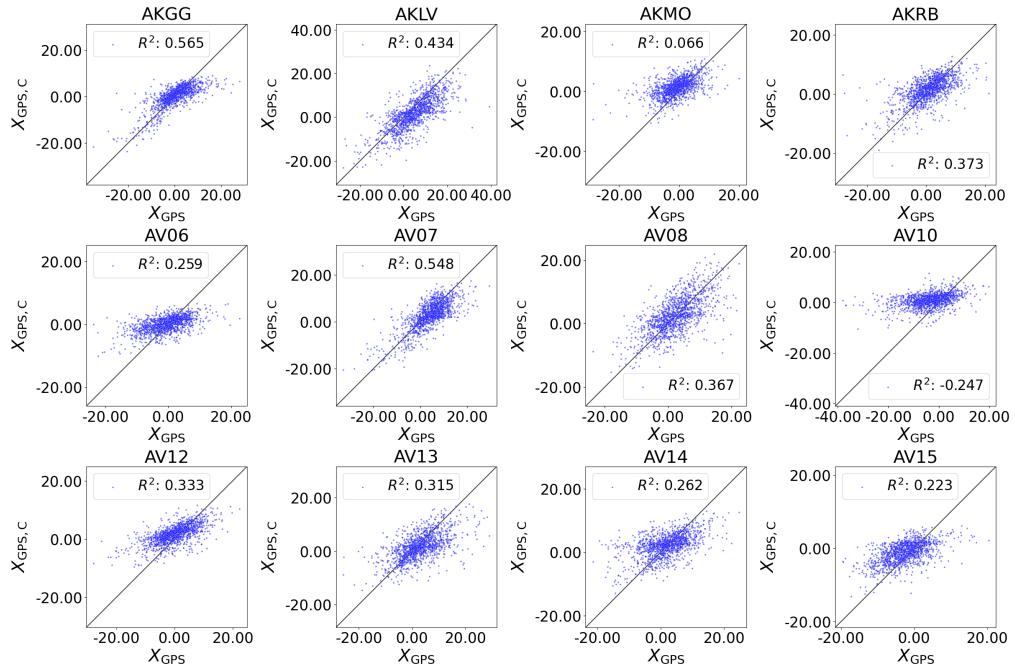
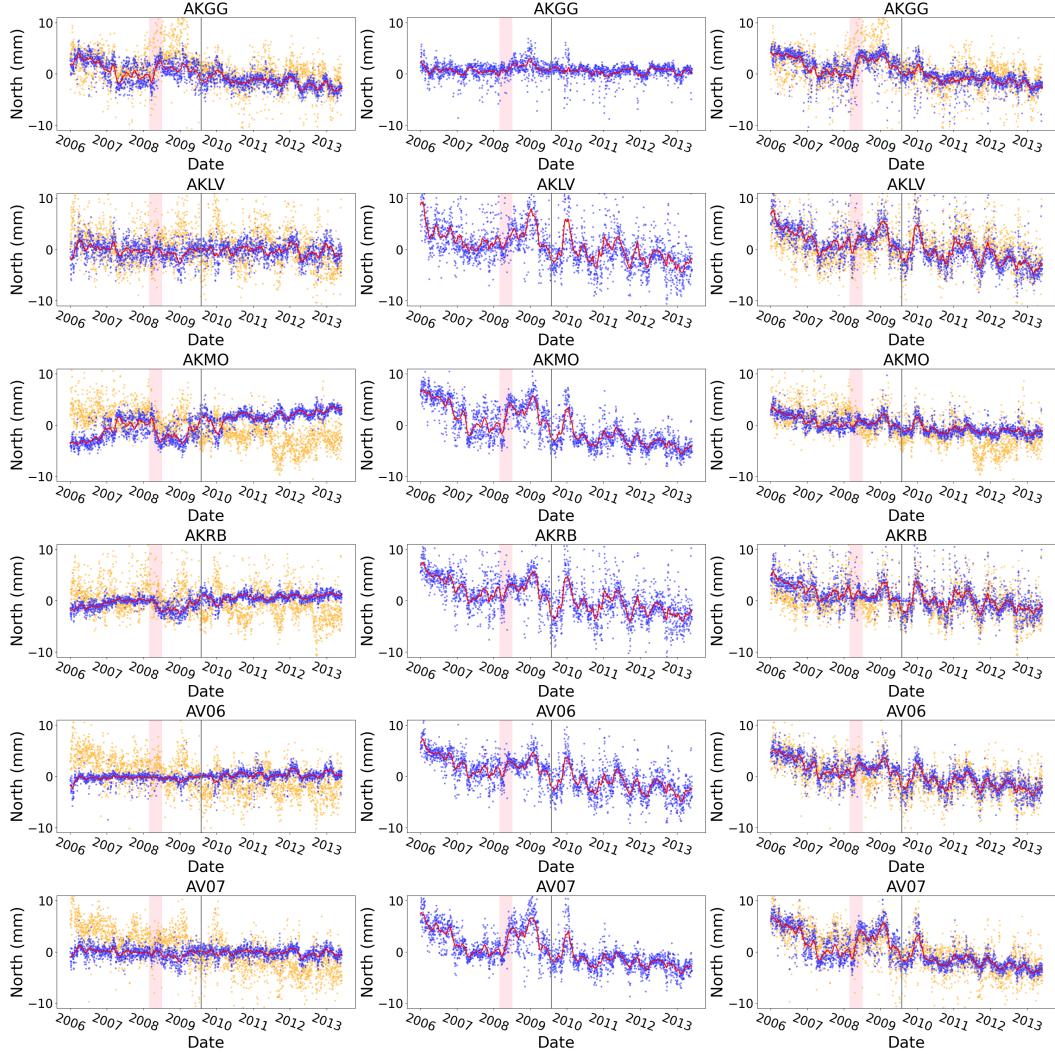
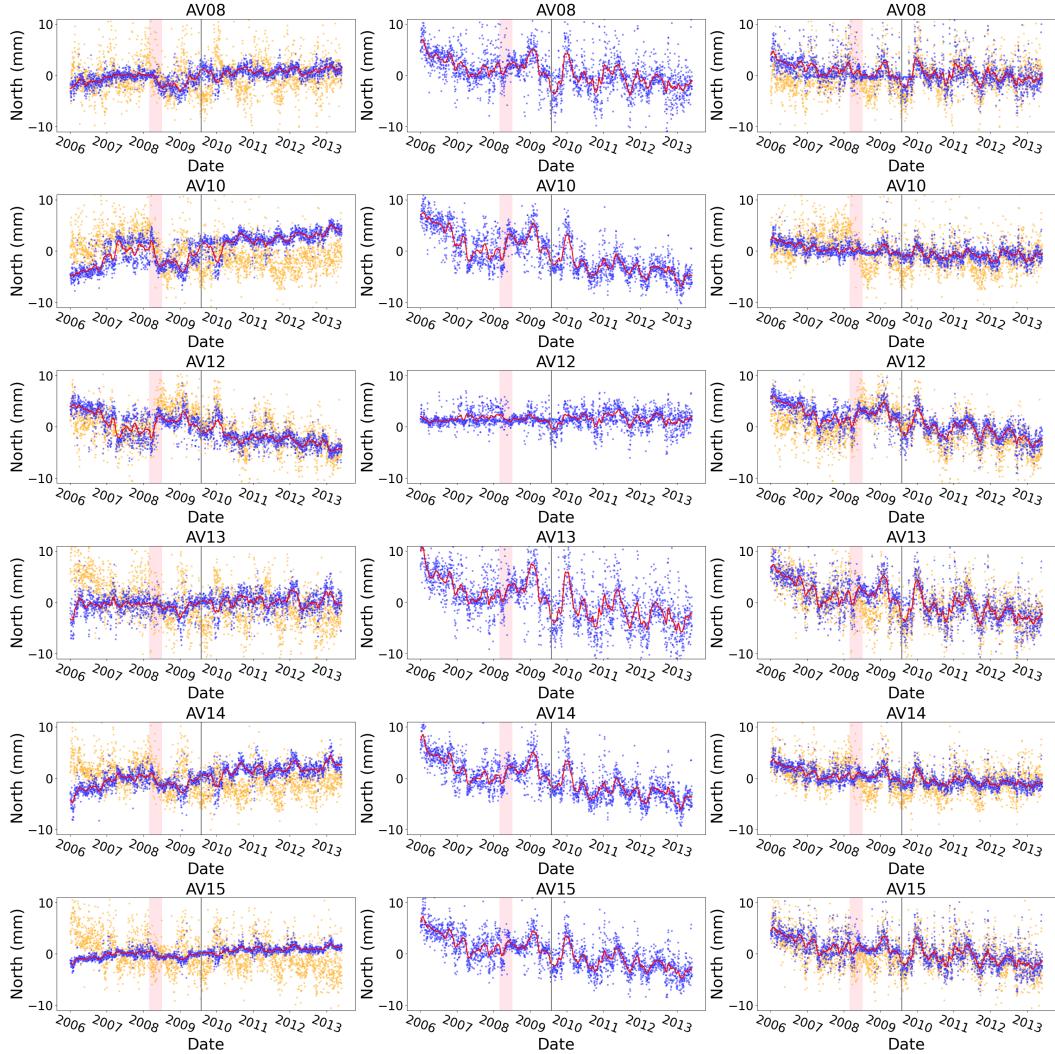


Figure 18: **Reconstruction accuracy of  $X_{C,\text{Mogi}}$  illustrated for vertical displacements.** The bias correction has improved the reconstruction but it is not as linear as for RTM, highlighting the trade-off between model simplicity and bias correction complexity.



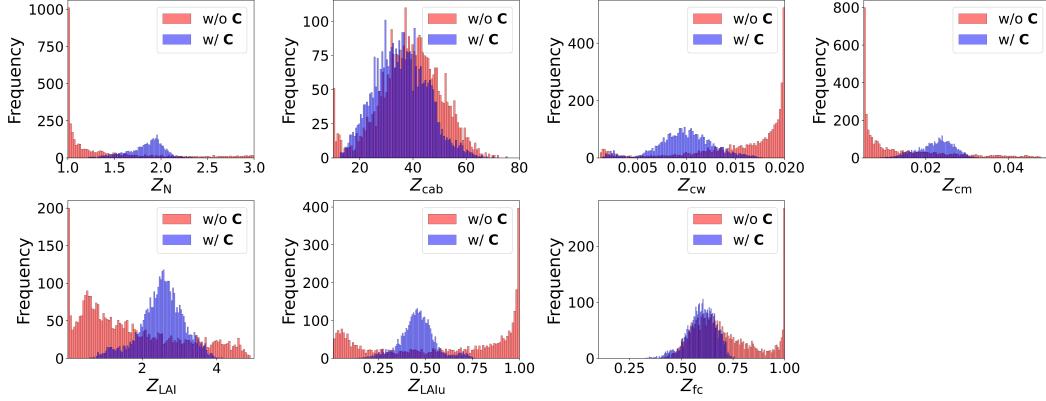
**Figure 19: Reconstructed GNSS signals and corrected biases illustrated for north displacements for the first half of 12 GNSS stations.** Orange: measured observations  $X_{S2}$ . Blue: Mogi output  $X_{F_{Mogi,C}}$  (first column); Corrected biases  $B(Q)_{C,Mogi} = X_{C,Mogi} - X_{F_{Mogi,C}}$  (second column); Corrected output  $X_{C,Mogi}$  (third column).  $X_{F_{Mogi,C}}$  clearly captures the transient signals in 2008, characterized by the radial extension along horizontal directions. In contrast,  $B(Q)_{C,Mogi}$  mainly captures the seasonal components and noises as suggested in eq. (14), and its temporal variations across stations do not contain prominent transient signals, which is aligned with our expectations about the seasonal components and random noises. On the left part of the black line are validations on the test set, and on the right part are validations on the training set. The predictions are consistent throughout the training and test sets.



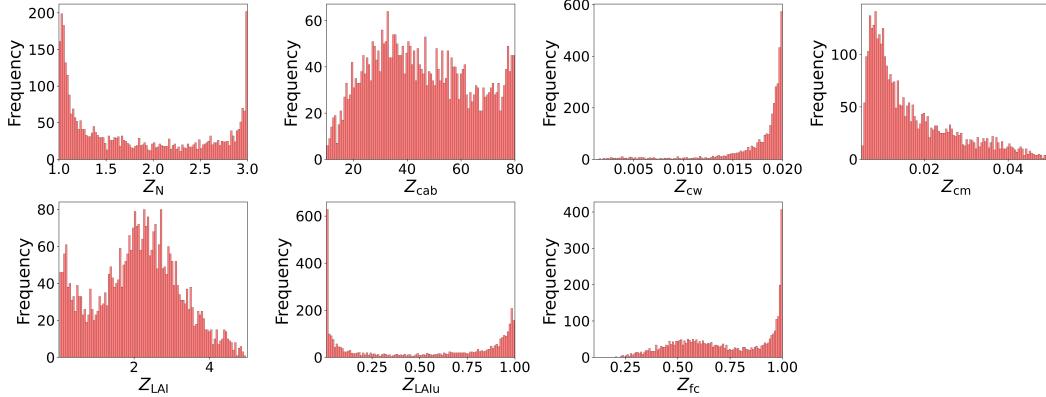
**Figure 20: Reconstructed GNSS signals and corrected biases** illustrated for north displacements for the second half of 12 GNSS stations. Orange: measured observations  $X_{S2}$ . Blue: Mogi output  $X_{F_{Mogi,C}}$  (first column); Corrected biases  $B(Q)_{C,Mogi} = X_{C,Mogi} - X_{F_{Mogi,C}}$  (second column); Corrected output  $X_{C,Mogi}$  (third column).  $X_{F_{Mogi,C}}$  clearly captures the transient signals in 2008, characterized by the radial extension along horizontal directions. In contrast,  $B(Q)_{C,Mogi}$  mainly captures the seasonal components and noises as suggested in eq. (14), and its temporal variations across stations do not contain prominent transient signals, which is aligned with our expectations about the seasonal components and random noises. On the left part of the black line are validations on the test set, and on the right part are validations on the training set. The predictions are consistent throughout the training and test sets.

## A.10 Extended results for the evaluation of physical variables

### A.10.1 Retrieved biophysical variables of $F_{RTM}$



**Figure 21: Distributions of variables are tightened up by bias correction** illustrated for  $Z_{C,RTM}$  vs.  $Z_{B,RTM}$ . Without bias correction the variable distributions (red) are implausible, tending to break out beyond the preset bounded ranges. With bias correction (blue), the model is able to learn more plausible distributions.



**Figure 22: Application of the regressor baseline  $M_{D,RTM}$  to  $X_{S2}$  leads to implausible parameter distributions of  $Z_{D,RTM}$  that tend to break out of the present parameter ranges.**

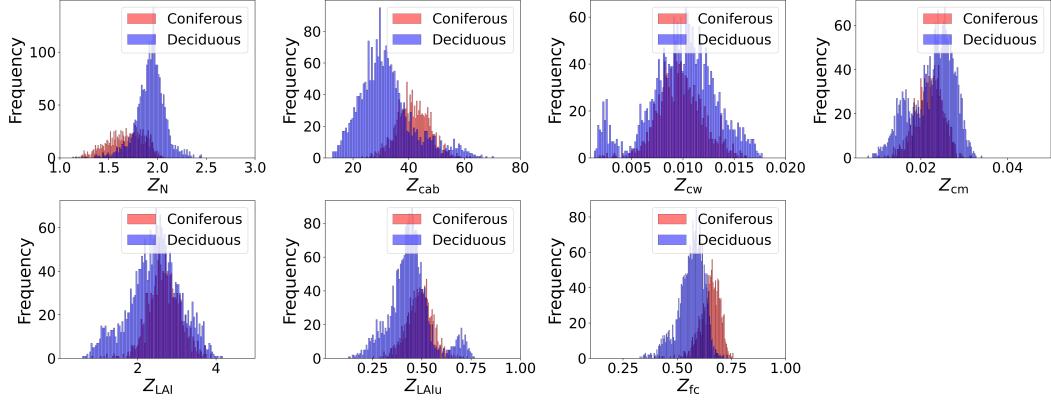


Figure 23:  $Z_{C,RTM}$  learned by our model are plausible and distinguish between forest types.

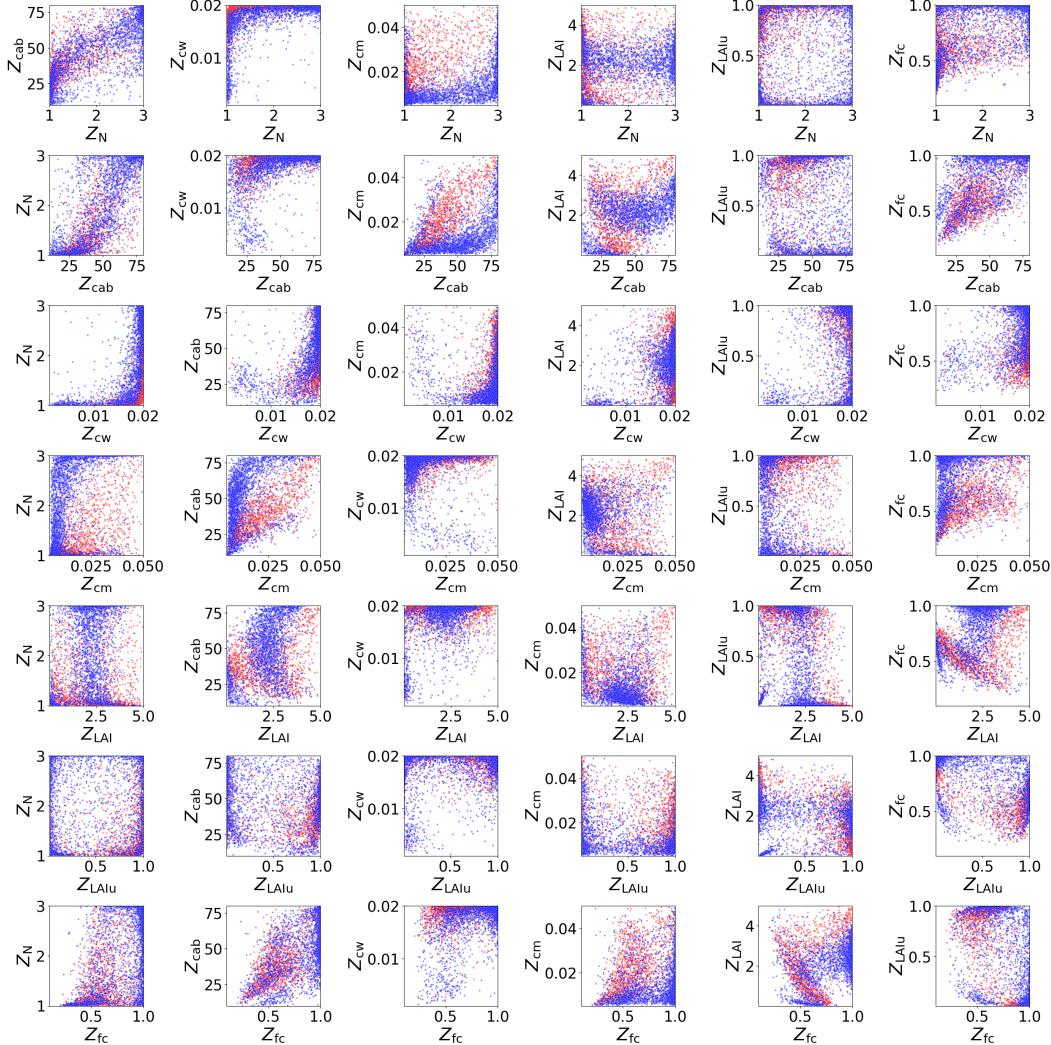


Figure 24: Pairwise co-distributions of  $Z_{D,RTM}$  learned by the regressor baseline show no clear physical patterns. Red: coniferous forest. Blue: deciduous forest.

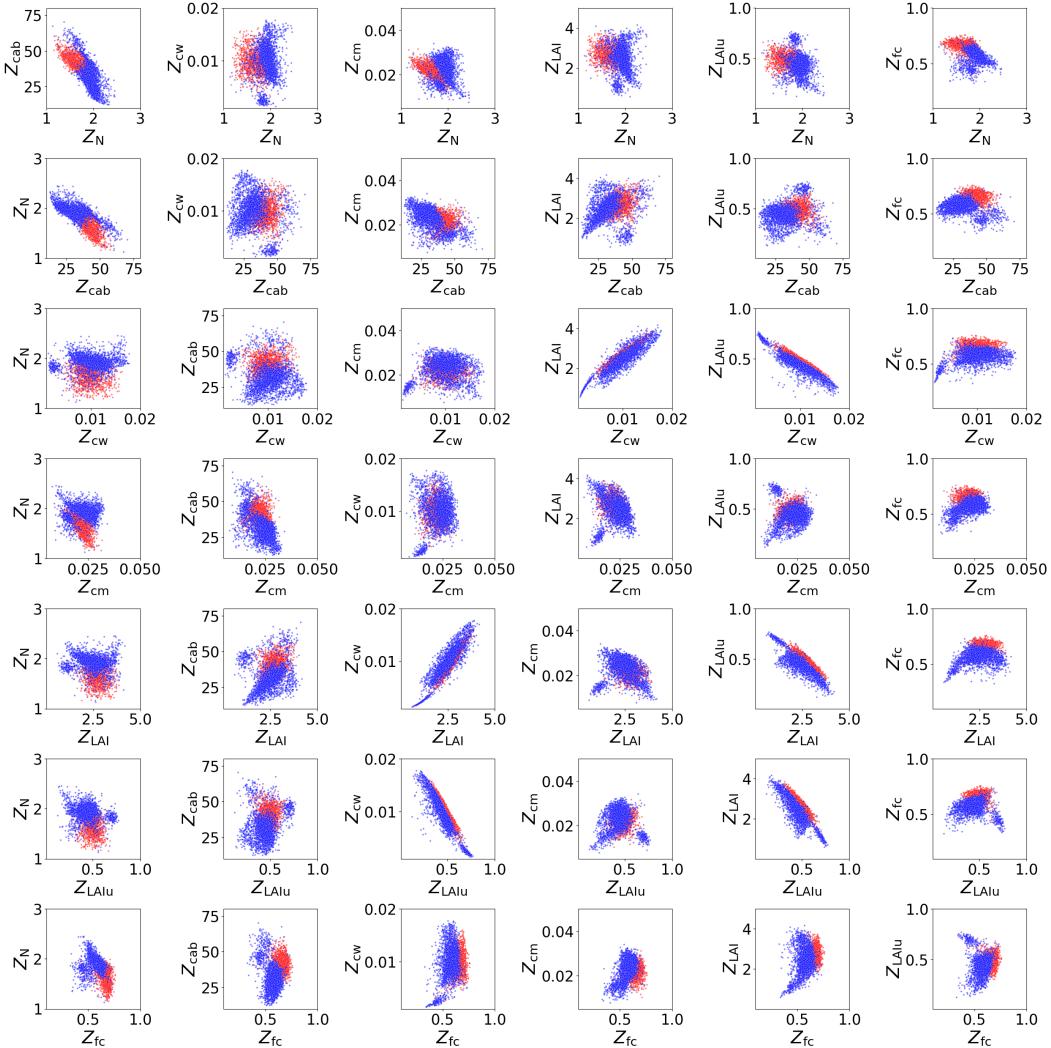


Figure 25: **Pairwise co-distributions of  $Z_{C,RTM}$  learned by our model show distinct and plausible physical patterns.** Red: coniferous forest. Blue: deciduous forest.

Table 9: **Statistics of  $Z_{D,RTM}$  learned by  $M_{D,RTM}$ .** We have also used  $Z_{D,RTM}$  to calculate the JM distance between species distributions, which are less disentangled compared to  $Z_{C,RTM}$ .

Species	N	cab	cw	cm	LAI	LAlu	fc
Pseudotsuga menziesii	$1.43 \pm 0.54$	$38.31 \pm 14.87$	$1.79 \pm 0.25$	$1.91 \pm 1.02$	$2.15 \pm 1.24$	$0.72 \pm 0.34$	$0.61 \pm 0.19$
Picea abies	$1.36 \pm 0.44$	$39.49 \pm 13.24$	$1.87 \pm 0.17$	$2.15 \pm 0.98$	$1.72 \pm 1.25$	$0.84 \pm 0.25$	$0.58 \pm 0.13$
Pinus nigra	$1.56 \pm 0.55$	$39.50 \pm 15.81$	$1.86 \pm 0.20$	$2.47 \pm 1.13$	$2.32 \pm 1.42$	$0.54 \pm 0.37$	$0.62 \pm 0.19$
Larix decidua	$1.50 \pm 0.50$	$38.26 \pm 13.85$	$1.61 \pm 0.42$	$1.43 \pm 0.69$	$1.90 \pm 1.04$	$0.56 \pm 0.40$	$0.67 \pm 0.21$
Pinus sylvestris	$1.56 \pm 0.58$	$42.34 \pm 15.68$	$1.90 \pm 0.11$	$1.99 \pm 1.01$	$2.40 \pm 1.25$	$0.67 \pm 0.37$	$0.59 \pm 0.18$
Prunus spp	$1.90 \pm 0.84$	$43.22 \pm 22.17$	$1.35 \pm 0.57$	$1.60 \pm 1.04$	$1.79 \pm 0.82$	$0.46 \pm 0.34$	$0.71 \pm 0.25$
Fagus sylvatica	$1.98 \pm 0.76$	$46.75 \pm 20.22$	$1.71 \pm 0.42$	$1.21 \pm 0.79$	$2.17 \pm 0.92$	$0.33 \pm 0.37$	$0.82 \pm 0.23$
Carpinus betulus	$1.96 \pm 0.71$	$47.55 \pm 19.40$	$1.67 \pm 0.36$	$1.36 \pm 0.84$	$1.91 \pm 0.91$	$0.55 \pm 0.40$	$0.75 \pm 0.21$
Quercus spp	$1.91 \pm 0.69$	$48.59 \pm 18.28$	$1.73 \pm 0.41$	$1.46 \pm 0.83$	$2.09 \pm 0.94$	$0.55 \pm 0.40$	$0.75 \pm 0.21$
Acer pseudoplatanus	$2.21 \pm 0.75$	$51.34 \pm 20.31$	$1.72 \pm 0.45$	$1.51 \pm 0.79$	$2.01 \pm 0.99$	$0.27 \pm 0.33$	$0.86 \pm 0.18$
Fraxinus excelsior	$2.16 \pm 0.71$	$51.67 \pm 18.05$	$1.76 \pm 0.37$	$1.93 \pm 1.17$	$2.05 \pm 1.07$	$0.44 \pm 0.39$	$0.78 \pm 0.21$
Alnus glutinosa	$2.22 \pm 0.72$	$55.01 \pm 18.09$	$1.78 \pm 0.34$	$2.03 \pm 0.97$	$2.04 \pm 1.03$	$0.46 \pm 0.41$	$0.76 \pm 0.20$

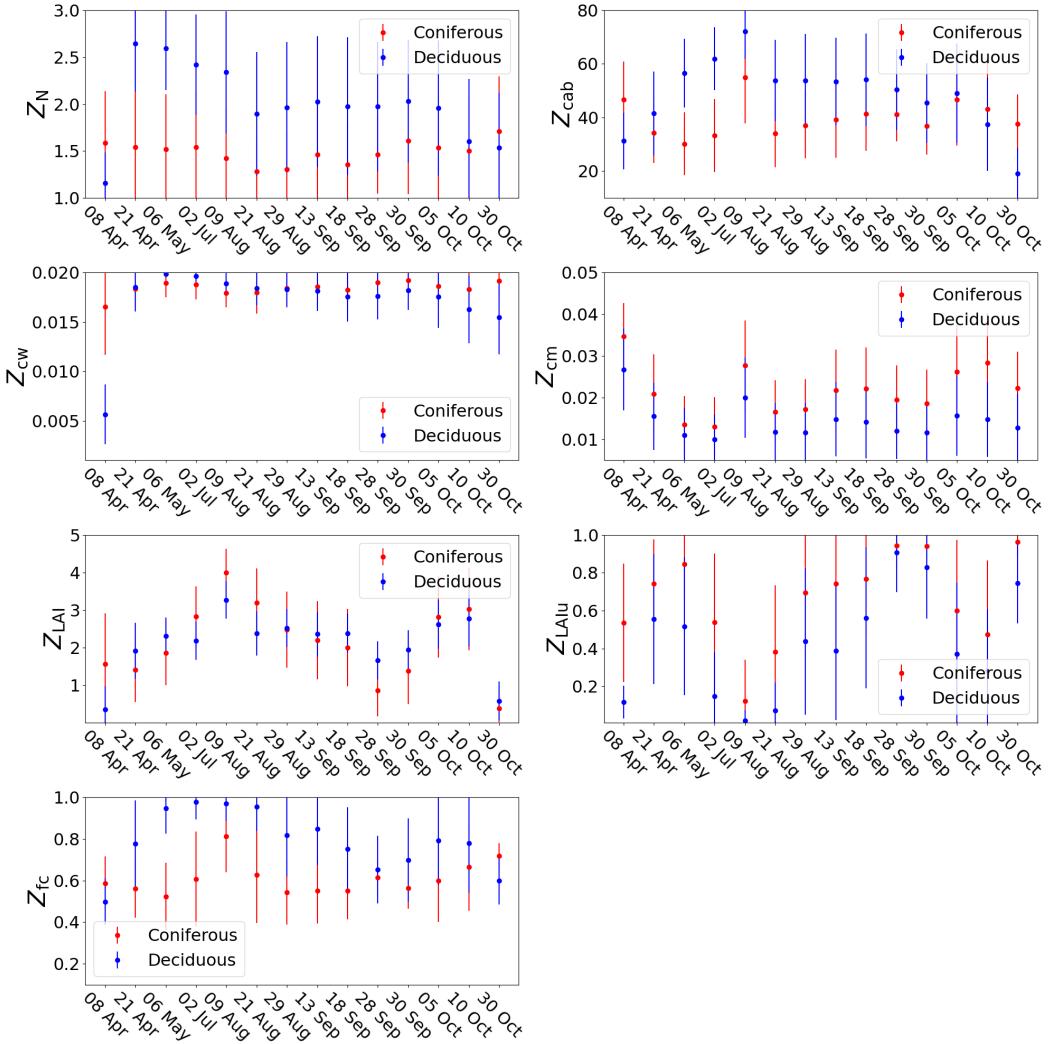


Figure 26: Temporal variations of inferred physical parameters  $Z_{D,RTM}$  are less consistent over time.

Table 10: Statistics of  $Z_{C,RTM}$  learned by  $M_{C,RTM}$ . We have also used  $Z_{C,RTM}$  to calculate the JM distance between species distributions and our model can learn disentangled representations.

Species	N	cab	cw	cm	LAI	LAiu	fc
Pseudotsuga menziesii	$1.65 \pm 0.16$	$39.96 \pm 5.33$	$1.00 \pm 0.24$	$2.36 \pm 0.23$	$2.66 \pm 0.43$	$0.49 \pm 0.07$	$0.66 \pm 0.04$
Picea abies	$1.53 \pm 0.16$	$43.33 \pm 4.89$	$0.99 \pm 0.24$	$2.37 \pm 0.21$	$2.70 \pm 0.43$	$0.50 \pm 0.07$	$0.68 \pm 0.04$
Pinus nigra	$1.75 \pm 0.16$	$42.71 \pm 5.63$	$0.96 \pm 0.19$	$1.97 \pm 0.28$	$2.73 \pm 0.37$	$0.48 \pm 0.06$	$0.64 \pm 0.04$
Larix decidua	$1.86 \pm 0.13$	$36.52 \pm 7.50$	$0.88 \pm 0.25$	$2.28 \pm 0.33$	$2.34 \pm 0.48$	$0.49 \pm 0.08$	$0.59 \pm 0.05$
Pinus sylvestris	$1.71 \pm 0.14$	$39.87 \pm 5.18$	$0.99 \pm 0.18$	$2.24 \pm 0.24$	$2.68 \pm 0.34$	$0.48 \pm 0.05$	$0.66 \pm 0.03$
Prunus spp	$1.93 \pm 0.12$	$34.71 \pm 8.65$	$0.80 \pm 0.30$	$2.26 \pm 0.37$	$2.13 \pm 0.54$	$0.50 \pm 0.10$	$0.53 \pm 0.05$
Fagus sylvatica	$1.98 \pm 0.17$	$29.76 \pm 10.71$	$0.94 \pm 0.33$	$2.24 \pm 0.53$	$2.32 \pm 0.69$	$0.43 \pm 0.11$	$0.54 \pm 0.06$
Carpinus betulus	$1.90 \pm 0.11$	$33.38 \pm 9.65$	$0.99 \pm 0.31$	$2.37 \pm 0.48$	$2.42 \pm 0.61$	$0.45 \pm 0.10$	$0.56 \pm 0.05$
Quercus spp	$1.87 \pm 0.11$	$33.46 \pm 8.55$	$1.06 \pm 0.33$	$2.35 \pm 0.48$	$2.56 \pm 0.63$	$0.44 \pm 0.10$	$0.59 \pm 0.06$
Acer pseudoplatanus	$1.99 \pm 0.14$	$30.82 \pm 10.09$	$0.99 \pm 0.34$	$2.12 \pm 0.49$	$2.45 \pm 0.68$	$0.42 \pm 0.12$	$0.54 \pm 0.05$
Fraxinus excelsior	$1.90 \pm 0.12$	$34.73 \pm 8.27$	$1.02 \pm 0.32$	$2.19 \pm 0.39$	$2.57 \pm 0.56$	$0.45 \pm 0.10$	$0.58 \pm 0.05$
Alnus glutinosa	$1.89 \pm 0.13$	$35.02 \pm 8.37$	$0.99 \pm 0.27$	$2.22 \pm 0.30$	$2.52 \pm 0.48$	$0.46 \pm 0.08$	$0.59 \pm 0.05$

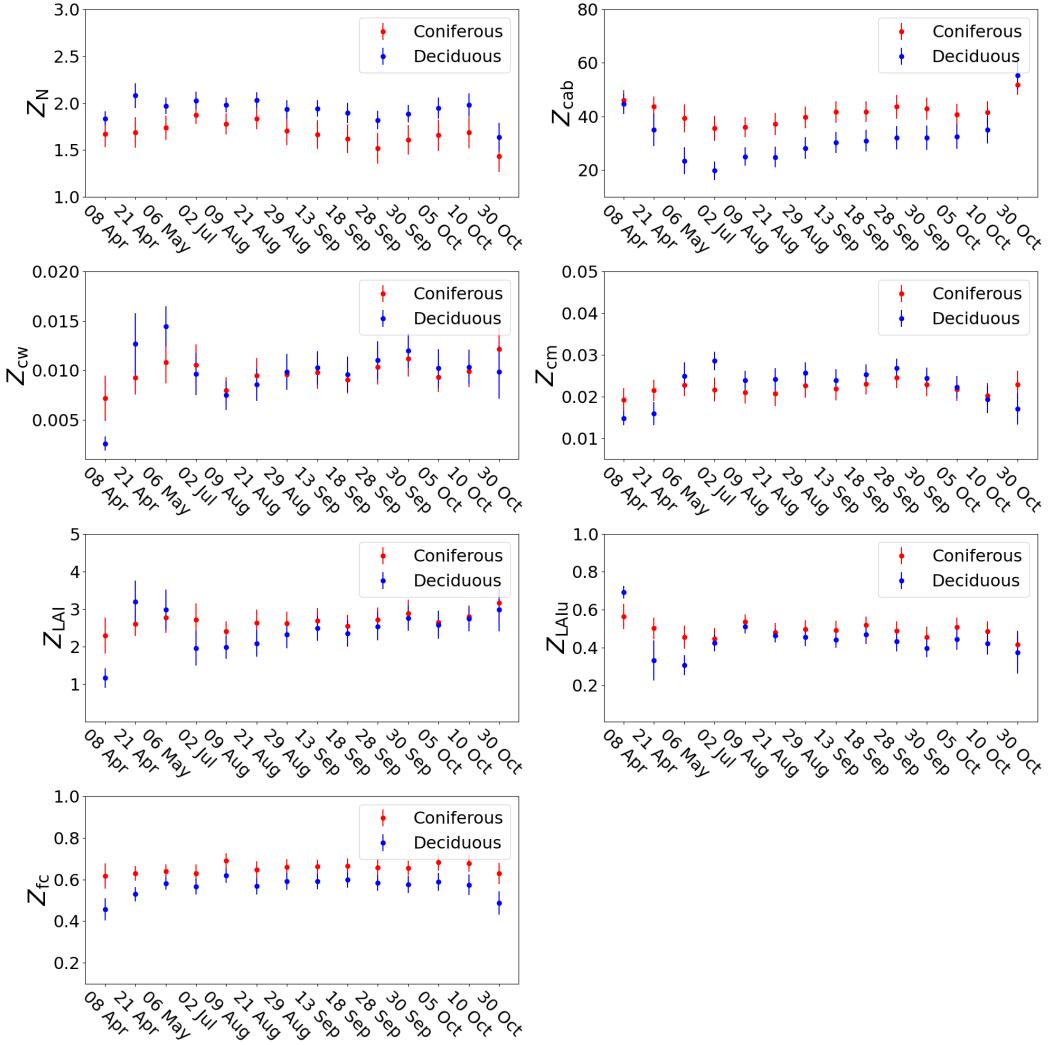
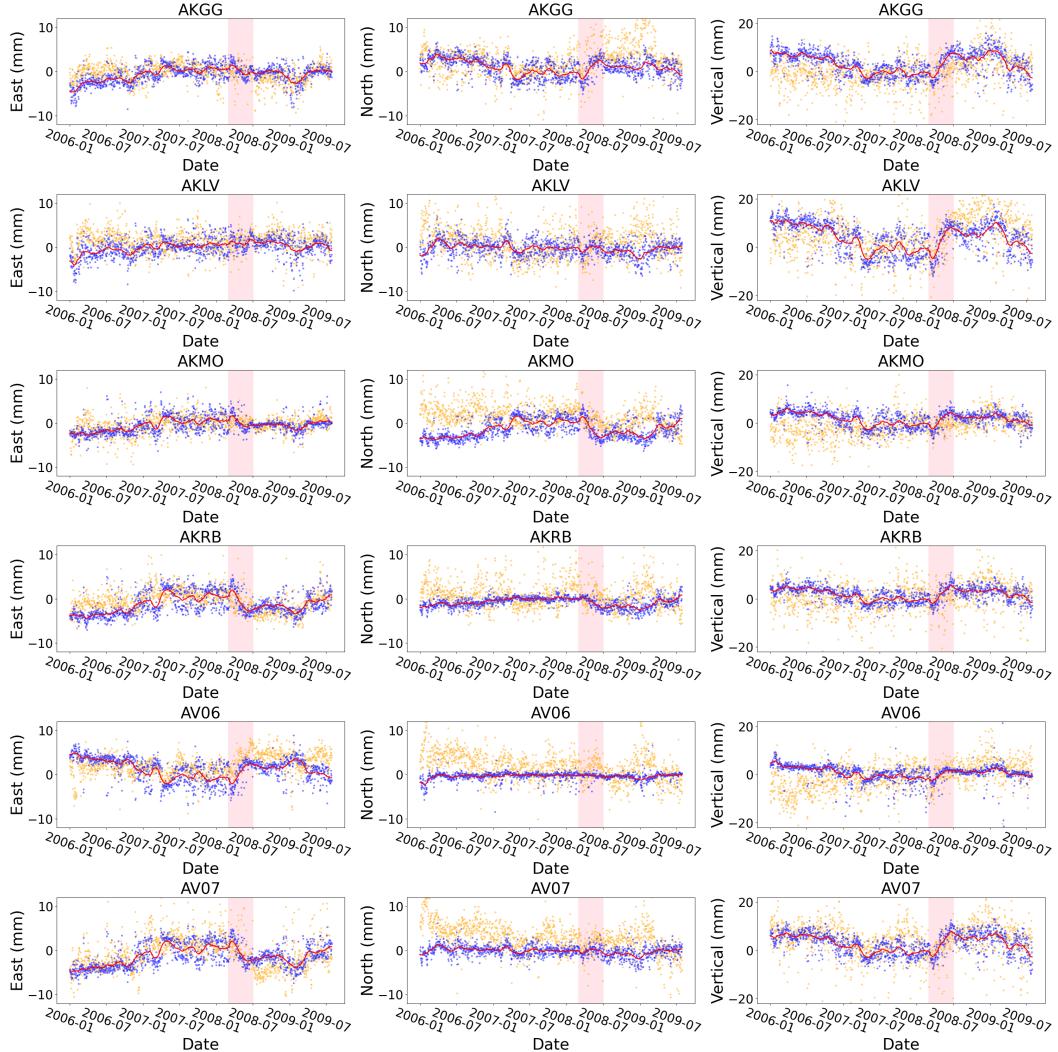
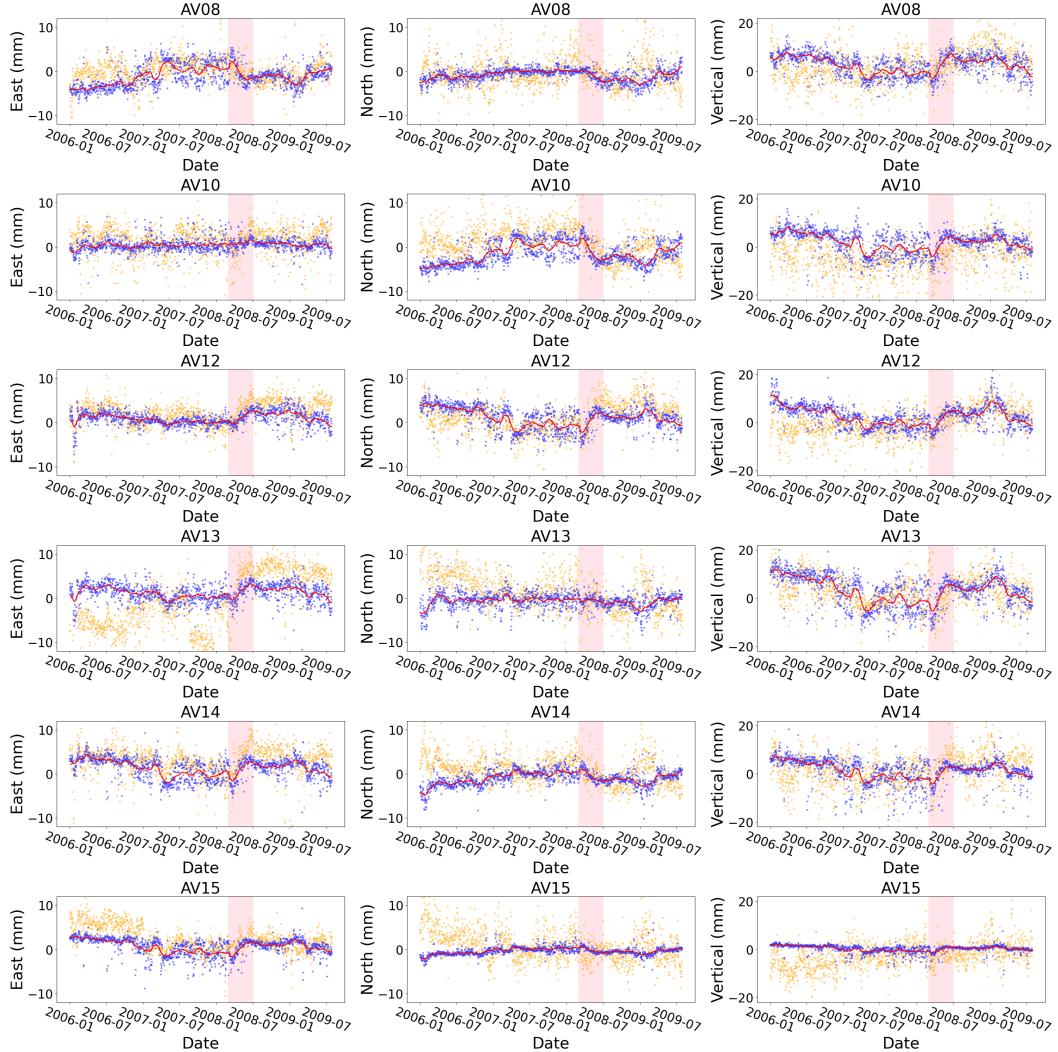


Figure 27: **Temporal variations of inferred physical parameters  $Z_{C,RTM}$ .  $M_{C,RTM}$  effectively captures distinct, temporally smooth and plausible variations for different forest types.**

#### A.10.2 Reconstructed GNSS signals based on $Z_{C,Mogi}$



**Figure 28: Time series of GNSS displacements** illustrated for the first half of 12 GNSS stations, located around the Mogi source. Orange:  $X_{\text{GNSS}}$ . Blue: Mogi reconstructions using  $Z_{\text{C},\text{Mogi}}$ . Red: Mogi reconstructions fitted by a Kalman filter. Our model successfully reconstructs the 2008 volcanic inflation transient signals, characterized by radial expansion and vertical uplift, with displacement magnitudes of 10 mm, aligned with prior findings [Ji and Herring, 2011, Walwer et al., 2016].



**Figure 29: Time series of GNSS displacements illustrated for the second half of 12 GNSS stations, located around the Mogi source. Orange:  $X_{\text{GNSS}}$ . Blue: Mogi reconstructions using  $Z_{C,\text{Mogi}}$ . Red: Mogi reconstructions fitted by a Kalman filter. Our model successfully reconstructs the 2008 volcanic inflation transient signals, characterized by radial expansion and vertical uplift, with displacement magnitudes of 10 mm, aligned with prior findings [Ji and Herring, 2011, Walwer et al., 2016].**