Dynamic Manifold Learning for Land Deformation Forecasting

Fan Zhou[1], **Rongfan Li**[1], **Qiang Gao**[2], **Goce Trajcevski**[3], **Kunpeng Zhang**[4], **Ting Zhong**[1]*
[1] University of Electronic Science and Technology of China
[2] Southwestern University of Finance and Economics
[3] Iowa State University
[4] University of Maryland, College park
fan.zhou@uestc.edu.cn, rongfanli1998@gmail.com, qianggao@swufe.edu.cn, gocet25@iastate.edu, kpzhang@umd.edu,
zhongting@uestc.edu.cn

Abstract

Landslides refer to occurrences of massive ground movements due to geological (and meteorological) factors, and can have disastrous impact on property, economy, and even lead to loss of life. The advances of remote sensing provide accurate and continuous terrain monitoring, enabling the study and analysis of land deformation which, in turn, can be used for possible landslides forecast. Prior studies either rely on independent observations for displacement prediction or model static land characteristics without considering the subtle interactions between different locations and the dynamic changes of the surface conditions. We present DyLand – *Dynamic Manifold Learning with Normalizing Flows for Land deformation prediction* – a novel framework for learning dynamic structures of terrain surface and improving the performance of land deformation prediction. DyLand models the spatial connections of InSAR measurements and estimates conditional distributions of deformations on the terrain manifold with a novel *normalizing flow*-based method. Instead of modeling the stable terrains, it incorporates surface permutations and captures the innate dynamics of the land surface while allowing for tractable likelihood estimates on the manifold. Our extensive evaluations on curated InSAR datasets from continuous monitoring of slopes prone to landslides show that DyLand outperforms existing bechmarking models.

Introduction

Landslides are among the most common catastrophic hazards occurring in various areas, resulting in fatalities and significant economic damages every year. They occur due to the gravity and/or elevation, and deformations of the terrain surface caused by numerous factors, such as water infiltration, glacier melting, aquifer exploitation, rock erosion, earthquakes, and volcanic eruptions (de Blasio 2020). The advanced geodetic techniques in remote sensing such as Interferometric Synthetic Aperture Radar (InSAR) and Global Navigation Satellite System (GNSS) enable accurate monitoring of slope deformations and forecasting the possibility of landslides, which have recently been studied (Carlà et al. 2019; Dong et al. 2019; Hajimoradlou, Roberti, and Poole 2020; Zhou et al. 2021).

Related work: Earlier works on landslide risk assessment largely relied on influential factors such as climate, rock stability, and slope gradient, as well as expert opinions from domain scientists (e.g., geologists and meteorologists) (Dai, Lee, and Ngai 2002; Wan, Lei, and Chou 2012; Thiery, Maquaire, and Fressard 2014; Zhu et al. 2014). These approaches require specific domain knowledge and are usually limited to *post hoc* explanations, and therefore is difficult to be generalized to places with different environmental conditions. More recent studies (Dong et al. 2019; Gan, Yang, and Zhou 2019; Carlà et al. 2019) have turned to modeling and prediction of landslides with machine learning (ML) techniques through leveraging the rich measurements of weather, surface, vegetation, and geology. Various ML models have been exploited in the literature, including analytical hierarchy process (AHP) (Kayastha, Dhital, and De Smedt 2013), Bayesian networks (Shirzadi et al. 2017), logistic regression (Kalantar et al. 2018), ensemble learning (Chen et al. 2017), and statistical learning (Hong et al. 2016).

The advances of neural networks have inspired several deep learning-based landslide prediction models. Convolutional Neural Networks (CNNs) are widely used for landslide susceptibility mapping (Lei et al. 2019; Hajimoradlou, Roberti, and Poole 2020), which generally rely on inventory mapping from the InSAR data to outline the landslide boundaries and deformation features. For example, LACNN (Hajimoradlou, Roberti, and Poole 2020) is a locally aligned CNN model that takes the orientation of each pixel at multiple scales to capture hidden features, and assess the landslide susceptibility at a specific point. SA-GNN (Zhou et al. 2021) embeds the 3D surface into a 2D graph with locally linear embedding (Roweis and Saul 2000), while preserving the relative positions and slopes of adjacent points on the surface. It then uses a Spatial-Temporal Graph Neural Network (ST-GNN) to capture the characteristics of neighboring points and predict the terrain deformations.

Challenges: Despite achieved promising results, existing land deformation models still confront several challenges. First, they either exploit 2D CNNs to extract the feature maps (Hajimoradlou, Roberti, and Poole 2020) or project the surface into a plain graph (Zhou et al. 2021) – inevitably introducing errors and, most importantly, ignoring vital information (e.g., azimuth, orientations, slopes) for assessing

*Corresponding author

the stability of the surface. Second, the InSAR observations, represented as point cloud data, are associated with temporal deformations beyond the geographical positions of the monitored areas. Therefore, directly applying methods for modeling point cloud or manifold (Wang et al. 2019; Shi and Rajkumar 2020) cannot capture the deformations of the surface. Lastly, prior studies are inadequate for probabilistic modeling and inference on the surface, which is essential for land stability estimation (inferring the distribution of deformations and understanding the mechanisms behind the deformations). While some recent works (Gemici, Rezende, and Mohamed 2016; Rezende et al. 2020; Kim et al. 2020; Brehmer and Cranmer 2020) estimate the distribution on the low-dimensional manifold, they focus on static surfaces. Thus, estimating the density on a dynamic manifold remains a challenge.

Present work: We propose a novel probabilistic manifold learning model called DyLand for terrain deformation prediction. Instead of projecting 3D point clouds to 2D space as previous methods (Hajimoradlou, Roberti, and Poole 2020; Shi and Rajkumar 2020; Zhou et al. 2021) did, we directly model the spatial structures on the manifold without important geometric information loss caused by dimensionality reduction. We also present a new generative model to simultaneously capture the topological dependencies and temporal deformations. DyLand considers the dynamics of surface and collective deformation trend in unison, rather than learning them separately, as in existing studies. As a principled dynamic manifold learning framework, DyLand generalizes normalizing flows (Papamakarios et al. 2021) for density estimation on the manifold surface while preserving the intricate dynamics of the continuously changed terrains. In sum, our contributions are three-fold:

- We explicitly explore the temporal dynamics of land surface and topological dependencies between monitored locations to study the land deformation prediction. We illustrate the conceptual limitations of existing methods and present a new framework to tackle two fundamental drawbacks by learning the dynamics of manifold and unifying the co-evolution of surface deformation and spatio-temporal representation.
- We propose a method for density estimation on a dynamic manifold that has not been studied before. Our method generalizes flow-based generative models to learn a probability density over the manifold, while preserving the dynamics via a local deformation perturbation strategy. This procedure not only enables the model to pay attention to the global dynamics of the surface beyond Euclidean positions, but also allows learning a conditional density that can substantially facilitate the deformation prediction.
- We consider the temporal deformation and spatio-temporal representation learning as a dynamic system, and introduce a method based on neural ordinary differentiable equations to unify the learning of spatial embedding and surface deformation. This design emphasizes the importance of terrain dynamics in the forecasts while better approximating the optimal posterior of land deformations.

We collected real-world InSAR point cloud data of slopes evolution and conducted extensive experiments to evaluate the effectiveness of our proposed model. The results show that DyLand significantly outperforms other state-of-the-art models in forecasting land deformation, learning dynamic manifold and providing interpretable predictions.

Preliminaries

Problem Definition: The InSAR point cloud data corresponds to a collection of N monitored locations, each with a unique $d = 3$ coordinates (longitude, latitude, elevation), represented as a vector of triplets $\mathbf{V} \in \mathbb{R}^{N\times d}$. Each location has an associated sequence of deformation observations of length T, denoted as $\mathbf{S} \in \mathbb{R}^{N\times T}$. The land deformations at τ-th timestamp are denoted as $\mathbf{S}^{\tau} \in \mathbb{R}^{N\times 1}$. The main goal of **land deformation prediction** is to forecast the surface displacement $\mathbf{Y} \in \mathbb{R}^{N\times T'}$ of all N locations in future timestamps T', which is a typical spatio-temporal prediction problem:

$$p(\mathbf{Y}|\mathbf{V}, \mathbf{S}). \tag{1}$$

Spatio-Temporal GNNs: To explicitly model the spatio-temporal dependencies between locations, GNNs are the widely used architectures. They introduce an adjacency matrix $\mathbf{A}$ to learn a spatio-temporal embeddings $\mathbf{W}$ of points $\mathbf{V}$, i.e., requiring an alternative implementation of Eq. (1) according to Bayes' rules:

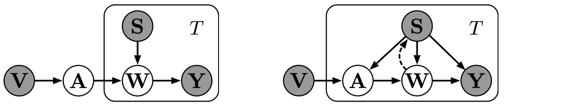
$$p(\mathbf{Y}, \mathbf{W}, \mathbf{A}|\mathbf{V}, \mathbf{S}) = p(\mathbf{A}|\mathbf{V}, \mathbf{S})p(\mathbf{Y}, \mathbf{W}|\mathbf{A}, \mathbf{V}, \mathbf{S}) \tag{2}$$

$$= \underbrace{p(\mathbf{A}|\mathbf{V}, \mathbf{S})}_{\textbf{Adjacency}} \underbrace{p(\mathbf{W}|\mathbf{A}, \mathbf{V}, \mathbf{S})}_{\textbf{GNN learning}} \underbrace{p(\mathbf{Y}|\mathbf{W}, \mathbf{A}, \mathbf{V}, \mathbf{S})}_{\textbf{Prediction}}, \tag{3}$$

where the first term $p(\mathbf{A}|\mathbf{V}, \mathbf{S})$ denotes the adjacency matrix construction; the second term $p(\mathbf{W}|\mathbf{A}, \mathbf{V}, \mathbf{S})$ is the spatio-temporal representation learning generally implemented by a specific GNN (e.g., GCN or GAT); and the last term is the future temporal prediction (e.g., traffic (Li et al. 2018) or land deformation in this work) that can be realized by any time series models, such as recurrent neural networks (Li et al. 2018; Wang et al. 2020).

Methodology: DyLand

The basic idea. In existing spatio-temporal GNNs (ST-GNNs), the adjacency matrix $\mathbf{A}$ can be directly built by $p(\mathbf{A}|\mathbf{V})$ as the locations $\mathbf{V}$ are generally fixed, e.g., the sensors in traffic forecasting (Li et al. 2018). This, however, may not fully capture the complex interactions between different locations, because the land surfaces are continuously deformed.



(a) Existing ST-GNN (b) DyLand

Figure 1: Graphical models of (a) the existing ST-GNNs and (b) our DyLand presented in this work.

In addition, the original spatio-temporal learning requires to model the joint distribution $p(\mathbf{Y}, \mathbf{W}|\mathbf{A}, \mathbf{V}, \mathbf{S})$ (cf.

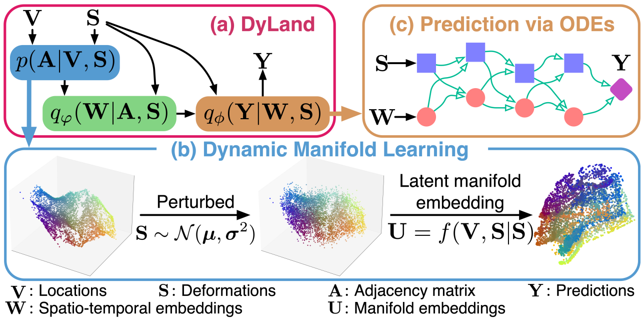

V
S
(a) DyLand
$p(\mathbf{A}|\mathbf{V},\mathbf{S})$
$\mathbf{Y}$
$q_\varphi(\mathbf{W}|\mathbf{A},\mathbf{S})$
$q_\phi(\mathbf{Y}|\mathbf{W},\mathbf{S})$
(c) Prediction via ODEs
S
W
Y
(b) Dynamic Manifold Learning
Perturbed
$\mathbf{S} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma}^2)$
Latent manifold embedding
$\mathbf{U} = f(\mathbf{V}, \mathbf{S}|\mathbf{S})$
V: Locations
S: Deformations
A: Adjacency matrix
Y: Predictions
W: Spatio-temporal embeddings
U: Manifold embeddings

Figure 2: Framework of the proposed DyLand. (a) The probabilistic learning of DyLand. (b) The dynamic manifold learning. (c) The surface deformation prediction with ODEs.

Eq.(2)). In existing works, $\mathbf{W}$ is usually learned from $\boldsymbol{\Omega}$ (here we let $\boldsymbol{\Omega} = \{\mathbf{A}, \mathbf{V}, \mathbf{S}\}$ for simplicity), and $\boldsymbol{\Omega}$ is therefore considered conditionally independent of $\mathbf{Y}$ (cf. Figure 1(a)), i.e., $\boldsymbol{\Omega} \perp\!\!\!\perp \mathbf{Y}|\mathbf{W}$. In this way, the spatio-temporal joint distribution is decomposed to a two-step learning as:

$$p(\mathbf{Y}, \mathbf{W}|\boldsymbol{\Omega}) = p(\mathbf{W}|\boldsymbol{\Omega})p(\mathbf{Y}|\mathbf{W}), \tag{4}$$

where the prediction $p(\mathbf{Y}|\mathbf{W})$ is made without considering the dynamics of nodes.

Overview of DyLand. Figure 2 illustrates the framework of DyLand. Specifically, we tackle two fundamental drawbacks of previous methods. **First**, we emphasize the importance of temporal features $\mathbf{S}$ beyond Euclidean positions in learning $p(\mathbf{A}|\mathbf{V}, \mathbf{S})$ through mapping both $\mathbf{S}$ and $\mathbf{V}$ into the manifold space, where the similarity among nodes and corresponding deformations can be better measured. Here, we propose a generative model based on normalizing flows (NFs) (Rezende et al. 2020; Brehmer and Cranmer 2020) to model $p(\mathbf{A}|\mathbf{V}, \mathbf{S})$ by learning the dynamics of the surface. **Second**, we model $\mathbf{S}$ and $\mathbf{W}$ jointly, i.e., $p(\mathbf{Y}|\mathbf{W}, \mathbf{S})$, rather than solely relying on $\mathbf{W}$ (e.g., $p(\mathbf{Y}|\mathbf{W})$) as in previous works. In this way, DyLand is able to better approximate the optimal prediction (cf. Eq.(3)), as $\mathbf{V}$ is given and the dynamics of $\mathbf{A}$ has been modeled in $p(\mathbf{A}|\mathbf{V}, \mathbf{S})$, as illustrated in Figure 1(b). Towards this goal, we consider the surface deformation as a dynamic system, and present an approach based on neural ordinary differential equations (ODEs) (Chen et al. 2018b) to predict and extrapolate the temporal deformations as $p(\mathbf{Y}|\mathbf{W}, \mathbf{S})$.

Dynamic Manifold Learning

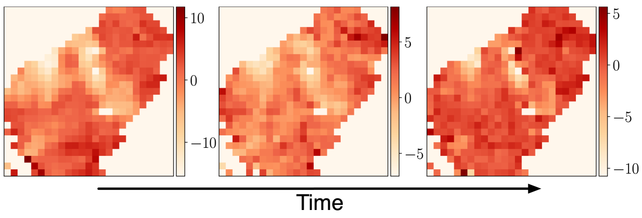


Figure 3: Deformation heat maps (real terrain).

As illustrated in Figure 3, neighboring areas on the surface exhibit similar displacement trend, which has inspired GNN-based models to aggregate the deformations from neighboring nodes (Zhou et al. 2021). However, the adjacency matrix $\mathbf{A}$ is static, ignoring the continuous displacement of the land surface. Besides, the influence of points with the same distance may not be consistent, because the relative positions (e.g., slope and azimuth) of neighboring points are of great importance for informing the trend of surface deformation (Hajimoradlou, Roberti, and Poole 2020; Zhou et al. 2021).

Towards that, we train a stochastic model to estimate the distribution of land deformations and model the dynamical adjacency according to the temporal evolution of $\mathbf{S}$. To encode the spatial correlations and mutual influence, we introduce a latent factor $\mathbf{u} \in \mathbf{U}$ and assume a deterministic mapping $p(\mathbf{A}|\mathbf{U})$ as:

$$p(\mathbf{A}|\mathbf{V}, \mathbf{S}) = \sum_{\mathbf{U}} p(\mathbf{A}|\mathbf{U})p(\mathbf{U}|\mathbf{V}, \mathbf{S}), \tag{5}$$

which is a dynamic adjacency generative model given the geographical locations of points $\mathbf{V}$ and their corresponding deformations $\mathbf{S}$.

At a specific time step $\tau \in [1, T]$, the latent embedding is estimated by a stochastic model f as $\mathbf{U}^\tau = f(\mathbf{V}^\tau, \mathbf{S}^\tau|\mathbf{S}^\tau)$. The basic idea is to generalize the geographical similarity and local deformation trend into the latent factors $\mathbf{U}$, and for the i-th location, the latent embedding $\mathbf{u}_i \in \mathbf{U}$ encodes the Euclidean spatial dependencies and the temporal deformations of adjacent areas.

Since the land surface is a typical manifold (i.e., 2D manifold data scattered over 3D space) and is continuously displaced, we need a technique to estimate the distribution of data points on the surface. Nevertheless, $\mathbf{V}$ is in Euclidean but the manifold is defined in non-Euclidean space in general, i.e., without affine properties. More importantly, we need a density estimation method for probability inference on the surface directly, instead of embedding the manifold into a 2D space that would inevitably result in information loss. A few efforts have studied how to learn a probability density over the manifold. For example, (Gemici, Rezende, and Mohamed 2016) first defines NF and estimate probability density on a Riemannian manifold embedded in the high-dimensional data space, and (Rezende et al. 2020; Brehmer and Cranmer 2020) investigates the expressiveness and stability of tractable density on the space that are not diffeomorphic to Euclidean such as tori and sphere. Unfortunately, these works define NF on a static manifold, which cannot be directly applied to model the dynamic surface.

Here we present an NF model to capture the interactions and dynamics of land surface through estimating the density on the manifold. Specifically, we add the deformation $\mathbf{s}_i^\tau$ to the $i-$th monitored location $\mathbf{v}_i$, and estimate the distribution of perturbed vector as $\mathbf{v}'_i = f^{-1}(\mathbf{u}_i^\tau|\mathbf{s}_i^\tau)$. Then we can exploit a NF as the generative model to learn the transformation from latent space $\mathbf{u}_i^\tau$ to the perturbed location $\mathbf{v}'_i$:

$$\log p(\mathbf{v}'_i|\mathbf{s}_i^\tau) = \log p(\mathbf{u}_i^\tau) - \log \left| \det \frac{\partial f^{-1}(\mathbf{u}_i^\tau|\mathbf{s}_i^\tau)}{\partial \mathbf{u}_i^\tau} \right|, \tag{6}$$

where a series of invertible mappings from uniform distribution $p(\mathbf{u}_i^\tau)$ is used to approximate the complex distribution $\log p(\mathbf{v}_i'|\mathbf{s}_i^\tau)$ conditioned on the deformation $\mathbf{s}_i^\tau$. There are many choices of the NF models (cf. (Papamakarios et al. 2021) for a comprehensive review). In our DyLand, we choose the continuous normalizing flows (Chen et al. 2018a; Grathwohl et al. 2018) as it imposes no restrictions on model architecture and the log-density follows the instantaneous change of variables formula, which can be described as:

$$\begin{aligned}\mathbf{u}^{t_1} &= \mathbf{u}^{t_0} + \int_{t_0}^{t_1} f^{-1}(\mathbf{u}^t, t, \mathbf{s}_i^\tau) dt, \\ \log p(\mathbf{u}^{t_1}|\mathbf{s}_i^\tau) &= \log p(\mathbf{u}^{t_0}) - \int_{t_0}^{t_1} \mathrm{Tr}\,\big(\frac{df^{-1}}{d\mathbf{u}^t}\big) dt,\end{aligned} \tag{7}$$

where $\mathbf{u}^{t_1} = \mathbf{v}_i'$ and $\mathbf{u}^{t_0} = \mathbf{u}_i^\tau$, and the trace can be efficiently computed by Hutchinson's estimator (Hutchinson 1989).

Note that the dimensions of input and output must be exactly the same as restricted by Eqs.(6) and (7). As a result, the transformations may be prevented at high dimensional space with lower information density embeddings. Recent studies (Kim et al. 2020; Dupont, Doucet, and Teh 2019) suggest to bridge the gap between different dimension scales by appending additional fully connected (FC) layers. Such dimension transformation is difficult when the dimensions of the original space are very high, but in our case, the transformation of 3D surface of $\mathbf{U}$ is feasible and easy to train.

Above we have exploited NF to capture the dynamics and spatial correlations of surface, i.e., $p(\mathbf{U}|\mathbf{V}, \mathbf{S})$. Then, for any two points with learned embeddings $\mathbf{u}_i$ and $\mathbf{u}_j$, the mapping $p(\mathbf{A}|\mathbf{U})$ can be directly computed as:

$$p(\mathbf{A}^\tau|\mathbf{U}^\tau) = \prod_{i=1}^{N}\prod_{j=1}^{N} p(A_{ij}^\tau|\mathbf{u}_i^\tau, \mathbf{u}_j^\tau), \tag{8}$$

$$\text{with} \quad p(A_{ij}^\tau = 0|\mathbf{u}_i^\tau, \mathbf{u}_j^\tau) = \mathrm{sigmoid}(\|\mathbf{u}_i^\tau - \mathbf{u}_j^\tau\|_2), \tag{9}$$

where $A_{ij}^\tau \in \mathbf{A}^\tau$ (similarity between the two points in the latent space is estimated with a logistic sigmoid function).

Unifying Deformation and Representation Learning

Recall that the spatio-temporal representation learning in Eq.(3) is usually learned by graph aggregations using, for example, a GNN – and we denote the representation as $\mathbf{W} \in \mathbb{R}^{N \times M}$. However, GNNs suffer from over-smoothing problem in general due to the underlying aggregation mechanisms especially when they are going deeper (Wu et al. 2020). Although (Zhao and Akoglu 2020) suggests to add an extra normalization layer to prevent node embeddings from becoming too similar, it may disturb the regular feature aggregation important for learning the trending of local deformations and thus deteriorates the prediction performance. Meanwhile, existing spatio-temporal GNNs (Zhou et al. 2021; Wu et al. 2020; Li et al. 2018; Wang et al. 2020) make temporal predictions based on $p(\mathbf{Y}|\mathbf{W})$, which implicitly assuming $\boldsymbol{\Omega} \perp\!\!\!\perp \mathbf{Y}|\mathbf{W}$ (cf. Eq.(4)). However, we argue that $\mathbf{S}$ is indispensable for making predictions.

Specifically, we consider the co-evolution of $\mathbf{S}$ and $\mathbf{W}$ as a dynamic system as:

$$\begin{bmatrix}\mathbf{S}^{t_1} \\ \mathbf{W}^{t_1}\end{bmatrix} = \begin{bmatrix}\mathbf{S}^{t_0} \\ \mathbf{W}^{t_0}\end{bmatrix} + \int_{t_0}^{t_1} g\left(\begin{bmatrix}\mathbf{S}^{t} \\ \mathbf{W}^{t}\end{bmatrix}, t\right) dt, \tag{10}$$

where $t_0 \in [1, T]$, $t_1 \in [T+1, T']$, and $\mathbf{S}^{t_1} = \widetilde{\mathbf{Y}}$ is exactly the prediction at time t_1. Function g is parameterized by a neural network to model the hidden dynamics. Note that the dynamic g is a homeomorphism, and the prediction preserve the topology of the $\mathbf{W}$. We solve the ODEs with the adaptive Runge–Kutta 4 (5) scheme of Dormand–Prince (Dormand and Prince 1980). Let $\mathbf{z}_0 = [\mathbf{S}^{t_0}, \mathbf{W}^{t_0}]^\intercal$, at the step $n+1$, the p-order approximation of $\mathbf{z}_{n+1}$ is given by:

$$\mathbf{z}_{n+1}^p = \mathbf{z}_n^p + h\sum_{i=1}^{l} b_i^p \mathbf{k}_i, \quad n = 0, 1, 2, \cdots, \tag{11}$$

$$\text{with} \quad \mathbf{k}_i = g\Big(\mathbf{z}_n^p + h\sum_{j=1}^{i-1} a_{i,j}\mathbf{k}_j, t_n + c_i h\Big), \tag{12}$$

where h is adaptive step size; vector $\mathbf{k}$ denotes the slopes of g; parameters l, a, b and c are all arranged in a mnemonic device – a.k.a. Butcher tableau. The local truncation error is on the order of $O(h^5)$, and is calculated as:

$$\mathbf{e}_{n+1} = \mathbf{z}_{n+1}^5 - \mathbf{z}_{n+1}^4 = h\sum_{i=1}^{l}\left(b_i^5 - b_i^4\right)\mathbf{k}_i, \tag{13}$$

which is used to tune h: given a maximum error $\mathbf{e}_{\text{max}}$, if $\mathbf{e}_{n+1} \geq \mathbf{e}_{\text{max}}$, we reduce the step size (e.g., $h = h/2$) to reduce error; otherwise, it increases h (e.g., $h \leftarrow 2h$) to accelerate the approximation. In other words, smaller $\mathbf{e}_{\text{max}}$ takes more steps n to compute the predictions.

Optimization & Training

Given historical T pairs of inputs $\mathbf{S}^\tau$ and $\mathbf{W}^\tau$, each of which has T' predictions made by the ODE-based model as presented in Eq.(10), we can obtain the final predictions $\widehat{\mathbf{Y}} \in \mathbb{R}^{N \times T'}$ by aggregating $\widetilde{\mathbf{Y}} \in \mathbb{R}^{N \times T \times T'}$ via additional FC layers. Besides, the deformations $\mathbf{Y}^\tau$ at a specific time should follow the prior distribution of deformations, i.e., a Gaussian $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma}^2)$. This goal can be attained via approximating the optimal $p(\mathbf{Y}^\tau|\mathbf{W}^\tau, \mathbf{S}^\tau)$ with a proposal distribution $q_\phi(\mathbf{Y}^\tau|\mathbf{W}^\tau, \mathbf{S}^\tau)$, which equals to minimize the following KL divergence:

$$\begin{aligned}&D_{\text{KL}}\left(q_\phi\left(\mathbf{Y}^\tau|\mathbf{W}^\tau, \boldsymbol{\Omega}^\tau\right)\|p\left(\mathbf{Y}^\tau|\mathbf{W}^\tau, \boldsymbol{\Omega}^\tau\right)\right) \\ \simeq &D_{\text{KL}}\left(q_\phi\left(\mathbf{Y}^\tau|\mathbf{W}^\tau, \mathbf{S}^\tau\right)\|p\left(\mathbf{Y}^\tau\right)\right) \\ = &-H[q_\phi\left(\mathbf{Y}^\tau|\mathbf{W}^\tau, \mathbf{S}^\tau\right)] - \mathbb{E}_{q_\phi}[\log p\left(\mathbf{Y}^\tau\right)],\end{aligned} \tag{14}$$

where H is the entropy and the expectation is estimated by Monte Carlo method. The first approximation holds because the mapping from $\mathbf{W}^\tau, \mathbf{S}^\tau$ to $\mathbf{Y}^\tau$ is a bijection. In implementation, the objective of Eq.(14) is implemented by aggregating the all predictions of all N monitored locations:

$$\begin{aligned}\mathcal{L} = &\frac{1}{N\Delta T}\sum_{\tau=T+1}^{T'}\sum_{i=1}^{N}(y_i^\tau - \hat{y}_i^\tau)^2 + \sum_{\tau=T+1}^{T'}\sum_{i=1}^{N} p\left(\hat{y}_i^\tau\right)\log p\left(\hat{y}_i^\tau\right) \\ &- \frac{1}{N}\sum_{\tau=T+1}^{T'}\sum_{i=1}^{N}\log\mathcal{N}\left(\hat{y}_i^\tau|\mu_i^\tau, {\sigma_i^\tau}^2\right),\end{aligned} \tag{15}$$

where $\Delta T = T' - T$ is the length of prediction, y_i^τ and $\hat{y}_i^\tau$ denote the true and predictions of i-th point at time τ, and the mean square error (MSE) of the deformations is used for training. We refer to Appendix A. and B. in the Supplementary Materials for more details on derivations and algorithms pertaining to this section.

Experiments

We now describe the experimental settings and our main observations. We note that additional experiments are available in Appendix C. and D. in the Supplementary Materials, and that, for reproducibility, the source code and datasets are also available in the Supplementary Materials.

Datasets: The studied areas used for experimental evaluations are two landslide-prone slopes in the southwest of Sichuan, China. We collect the InSAR measurements of the monitored slopes, recording surface deformations over time. The **HZY** data is composed of the observations of the slopes on both sides (East and West) of the Dadu River, between Nov 30, 2018, to Sep 8, 2019. The **PBG** data has more than two years of InSAR measurements (Nov 17, 2017 – Jan 04, 2020) of both sides of the river. Table 1 summarizes the statistics of the two study areas.

Dataset	**HZY-West**	**HZY-East**
Time	11/30/2018 – 09/08/2019	
# nodes	4,569	2,164
Longitude (E102°)	[1’50”, 3’2”]	[2’35”, 3’46”]
Latitude (N30°)	[39’12”, 41’25”]	[39’38”, 40’48”]
Elevation	[1671.2, 2527.4]	[1470.2, 2899.6]
Displacement	[-27.58, 28.03]	[-29.06, 30.50]
	PBG-West	**PBG-East**
Time	11/17/2017 – 01/04/2020	
# nodes	5,886	8,671
Longitude (E102°)	[46’36”, 48’11”]	[52’31”, 54’17”]
Latitude (N29°)	[14’26”, 15’50”]	[12’41”, 14’3”]
Elevation	[1013.8, 2101.0]	[661.3, 1907.8]
Displacement	[-60.45, 110.57]	[-58.87, 49.51]

Table 1: Descriptive statistics of datasets.

Baselines: We compare DyLand with the following approaches: (1) Historical Average (HA). It calculates the historical data average of time period T as basic predictive capabilities for contrast. We have $T = 1$ and $T = 3$ marked as HA(1) and HA(3) respectively. (2) SVR is a regression version of support-vector machine. (3) Autoregressive Integrated Moving Average model (ARIMA), combines autoregressive and moving average for prediction. (4) GRU (Cho et al. 2014) captures the long-short term dependency which has been wildly used for time-series forecasting. (5) NODE (Chen et al. 2018b) is a continuous normalizing flow model that learns the time series by solving the ordinary differential equations. (6) GCN (Kipf and Welling 2017) models spatial dependencies via graph convolution. (7) VGAE (Kipf and Welling 2016) is an unsupervised variational framework making use of latent variables learned from the GCN autoencoder. (8) SIG-VAE (Hasanzadeh et al. 2019) is similar to VGAE but has more flexible and complex posterior approximation. (9) STGCN (Wu et al. 2020) filters inputs and hidden states passed to a recurrent unit using graph convolutions. (10)P-GNN (Shi and Rajkumar 2020) is a point cloud graph convolution method, and we adapt it to deformation prediction by adding GRUs. (11)SA-GNN (Zhou et al. 2021) learns the embeddings of the surface manifold and uses GCNs and GRUs to model spatio-temporal dependencies.

Settings, parameters & metrics: We split all datasets into three parts: 50% for training, 30% for validation, and the remaining 20% for testing. Geographical coordinates are max-min normalized. All deep learning methods are optimized by Adam optimizer (Kingma and Ba 2017) with learning rate of 10^{-3} and weight decay of 10^{-5}. Early stop is triggered when the loss has not declined for 100 consecutive epochs. All experimental results are the best in 20 runs without other specified and the statistical significance of the results is less than 0.05, i.e., $p < 0.05$.

In our DyLand, the latent manifold $\mathbf{U}$ is 3D for evaluation and 2D for visualization. The dimension of spatio-temporal representation $\mathbf{W}$ is 5. The deformation for each location $\mathbf{s}_i^\tau$ is scaled up by multiplying 10 and then passed to NF for conditioning, since the scale of $\mathbf{s}_i^\tau$ is too small compared to $\mathbf{V}$. The NF and neural ODEs have 3 layers and 64 dimensions, while both FC and GNN have 3 layers.

We use five standard evaluation metrics for the model performance: RMSE, MAE, ACC, R^2 and EVS (cf. Appendix C).

Performance Comparisons

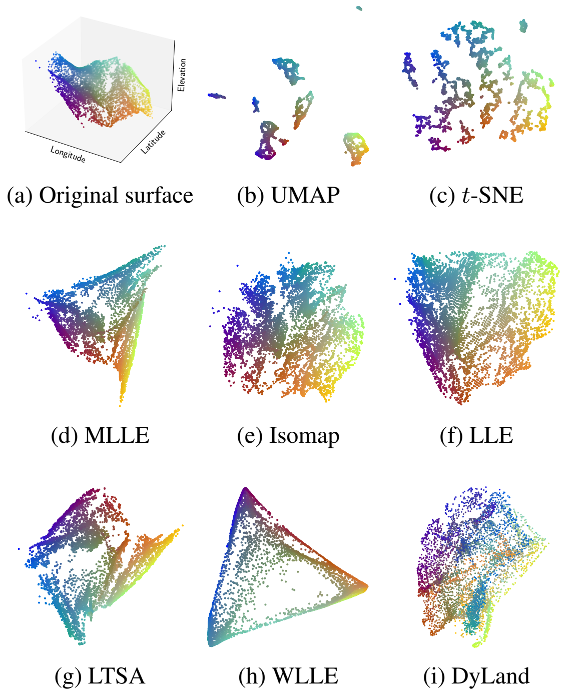
Table 2 reports the performance of different models on deformation prediction, where the best are in bold. First, models simply relying on temporal observations (e.g., HA, SVR, ARIMA, GRU, and NODE) is not enough to predict the land deformation as they ignore important spatial correlations of the monitored locations. In contrast, GNN-based approaches generally achieve better results, which demonstrates that modeling spatial relations and collective deformation trends can improve prediction performance. When comparing different GNN-based models, we can find that P-GNN and SA-GNN perform better than other spatial and temporal GNNs defined on the Euclidean space due to the ability to capture the complex correlations on the surface manifold. SA-GNN slightly outperforms P-GNN because the relative positions such as slope and azimuth are explicitly embedded, which can partially discriminate the importance of adjacent points.

Finally, our DyLand shows superior performance than all other models in terms of all metrics on four datasets. Compared to the best baseline model SA-GNN, DyLand is capable of modeling manifold dynamics instead of static surface embedding in SA-GNN, thus alleviating the bias caused by the surface deformation. Besides, DyLand considers the deformation learning and spatio-temporal embedding as a dynamic system that bridges the gap between the separate steps modeled in previous GNN-models – as a result, it is more accurate for deformation prediction.

Why Modeling Dynamic Manifold? Dynamic manifold learning plays an important role in our DyLand. To test its effect, we replace it with representative mani-

Method	HZY-West					HZY-East					PBG-West					PBG-East				
	RMSE	MAE	ACC	R^2	EVS	RMSE	MAE	ACC	R^2	EVS	RMSE	MAE	ACC	R^2	EVS	RMSE	MAE	ACC	R^2	EVS
HA(1)	4.190	2.799	0.072	0.054	0.161	4.973	3.427	0.056	0.065	0.178	8.348	5.114	0.044	0.062	0.120	6.944	4.194	0.046	0.061	0.117
HA(3)	3.992	2.889	0.052	0.066	0.122	4.645	3.352	0.045	0.094	0.158	7.535	5.186	0.040	0.092	0.142	6.067	4.010	0.050	0.097	0.164
SVR	3.956	3.209	0.030	0.108	0.274	4.750	3.693	0.032	0.135	0.262	6.612	4.199	0.059	0.195	0.291	5.955	3.546	0.076	0.191	0.282
ARIMA	4.888	4.088	0.053	0.265	0.215	6.704	5.395	0.031	0.271	0.204	6.242	4.098	0.035	0.190	0.212	5.325	3.200	0.055	0.175	0.228
GRU	0.250	0.204	0.421	0.223	0.315	0.197	0.170	0.456	0.207	0.290	0.249	0.197	0.460	0.120	0.163	0.200	0.160	0.540	0.215	0.137
NODE	0.076	0.057	0.527	0.407	0.458	0.101	0.075	0.506	0.469	0.417	0.052	0.044	0.584	0.283	0.291	0.053	0.041	0.710	0.401	0.412
GCN	0.100	0.076	0.442	0.365	0.332	0.103	0.078	0.440	0.399	0.343	0.063	0.050	0.574	0.204	0.208	0.089	0.056	0.662	0.385	0.345
VGAE	0.091	0.068	0.483	0.390	0.391	0.104	0.077	0.465	0.392	0.313	0.051	0.040	0.601	0.375	0.332	0.083	0.052	0.712	0.355	0.348
SIG-VAE	0.088	0.065	0.525	0.374	0.422	0.096	0.071	0.513	0.499	0.440	0.045	0.037	0.734	0.405	0.426	0.079	0.050	0.775	0.408	0.393
STGCN	0.069	0.055	0.533	0.456	0.451	0.078	0.057	0.562	0.563	0.573	0.040	0.029	0.815	0.447	0.488	0.041	0.027	0.854	0.437	0.426
P-GNN	0.065	0.048	0.628	0.423	0.466	0.071	0.051	0.637	0.507	0.516	0.031	0.020	0.908	0.474	0.491	0.032	0.026	0.911	0.441	0.449
SA-GNN	0.058	0.037	0.718	0.488	0.483	0.062	0.048	0.698	0.497	0.501	0.021	0.015	0.964	0.492	0.480	0.024	0.018	0.956	0.470	0.478
DyLand	**0.052**	**0.035**	**0.742**	**0.492**	**0.492**	**0.060**	**0.043**	**0.700**	**0.583**	**0.590**	**0.013**	**0.010**	**0.993**	**0.538**	**0.540**	**0.016**	**0.013**	**0.978**	**0.487**	**0.496**

Table 2: Overall performance comparisons on land deformation prediction.



(a) Original surface (b) UMAP (c) *t*-SNE

(d) MLLE (e) Isomap (f) LLE

(g) LTSA (h) WLLE (i) DyLand

Figure 4: Learned latent manifolds on PBG-West.

fold learning methods, including LLE (Roweis and Saul 2000), MLLE (Zhang and Wang 2007), HLLE (Donoho and Grimes 2003), LTSA (Zhang and Zha 2004), *t*-SNE (Maaten and Hinton 2008), WLLE (Zhou et al. 2021), Isomap (Tenenbaum, De Silva, and Langford 2000), and UMAP (McInnes, Healy, and Melville 2020).

Recall that DyLand models $p(\mathbf{A}|\mathbf{V},\mathbf{S})$ which explicitly considers the temporal deformation $\mathbf{S}$. To ensure a fair comparison, we take $\mathbf{S}$ as another temporal channel in addition to the spatial 3D channels for other models. Figure 4 plots the 2D representations learned by different models, colored by their geographical coordinates in original space. As typical dimension reduction algorithms, UMAP and *t*-SNE generally obtain clustered and scattered embeddings, and therefore lose continuous spatial correlations essential for deformation prediction. Other static manifold learning methods such as MLLE, Isomap, LLE, and LTSA can preserve the original manifold due to their ability of capturing local Euclidean topology. However, they model the static manifold in general and cannot discriminate the influence of neighboring nodes. WLLE, in contrast, takes the slope and azimuth information into account, but it significantly distorts the latent representations where local manifold structure is disregarded.

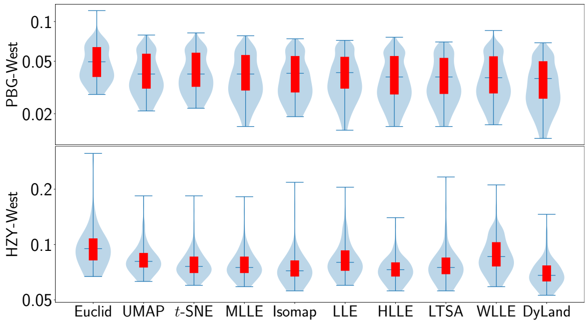
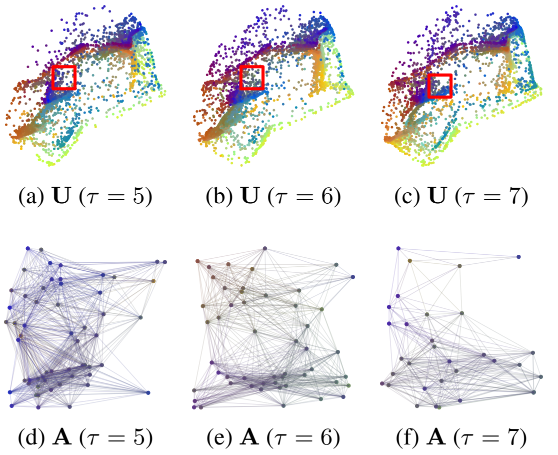


Figure 5: Quantitative comparisons between dynamic manifold and static manifold learning ("Euclid" takes the original surface as input directly; red lines present the quartiles).

DyLand overcomes the above issues by jointly preserving local manifold structure and temporal evolution of deformations. To quantify the benefits, we compare its prediction results with the variants with alternative manifold learning, and show them as violin plot in Figure 5. Clearly, DyLand outperforms other manifold learning methods which validates our motivation of modeling dynamic surface in land deformation prediction. Although the local topology patterns on the point cloud surface can be preserved in other manifold-based variants, land deformations are not explicitly considered, resulting in fixed adjacency and thus determined neighboring feature aggregation (as they cannot capture the intricate correlations beyond Euclidean space).


(a) $\mathbf{U}$ $(\tau = 5)$ (b) $\mathbf{U}$ $(\tau = 6)$ (c) $\mathbf{U}$ $(\tau = 7)$

(d) $\mathbf{A}$ $(\tau = 5)$ (e) $\mathbf{A}$ $(\tau = 6)$ (f) $\mathbf{A}$ $(\tau = 7)$

Figure 6: Comparisons of manifold and sampled relevant adjacency matrix at different time stamps on PBG-West.

To understand the dynamic adjacency modeled in DyLand intuitively, we show the manifold embeddings at different time stamps in Figure 6, as well as the adjacency matrices of the sampled nodes (framed by the red rectangle in top row). The results illustrate how embeddings and corresponding adjacency are changing over time, which is more obvious when observing adjacency matrices. Since DyLand can learn dynamic manifold from deformations and construct the adjacency matrices (i.e., interactions among nodes) accordingly, this property allows us to capture the interactions of nodes adaptively without the efforts of hyperparameter tuning. In contrast, previous static manifold learning approaches are parameter sensitive, e.g., presumed near neighbor specification. More importantly, DyLand generates adjacency $\mathbf{A}$ considering both geographical locations $\mathbf{V}$ and temporal features $\mathbf{S}$, enabling the model to aggregate the deformations of locations dynamically for adaptive prediction.

Effect of Co-Training $\mathbf{W}$ and $\mathbf{S}$: Recall that we consider the evolution of $\mathbf{W}$ and $\mathbf{S}$ as a dynamic system solved by a neural ODEs, where $\mathbf{W} \in \mathbb{R}^{N \times M}$ is the aggregated spatio-temporal features output by a GNN. Now we investigate the effect of co-training and its influence on the final prediction. We generate two variants of DyLand : (1) DyLand-$\mathbf{S}$, which removes the learning of $\mathbf{W}$ and makes prediction based on the historical observations – i.e., it degenerates to a typical ODE-based time series method (Chen et al. 2018b) through modeling $p(\mathbf{Y}|\mathbf{S})$; and (2) DyLand-$\mathbf{W}$, which only utilizes the learned spatio-temporal representation to make prediction, i.e., $p(\mathbf{Y}|\mathbf{W})$, as most existing spatio-temporal GNNs. As summarized in Table 3, modeling the co-evolution of $\mathbf{S}$ and $\mathbf{W}$ as a holistic dynamic system significantly improves the performance. DyLand-$\mathbf{S}$ performs worst due to ignoring the spatial dependencies, although it also uses neural ODEs to learn the dynamic system.

Co-training $\mathbf{S}$ and $\mathbf{W}$ with neural ODEs can be considered as an augmentation to the dynamic system of $\mathbf{S}$, since representation $\mathbf{W}$ is also conditioned on the evolution of deformations $\mathbf{S}$. In other words, it solves the ODEs regarding $\mathbf{S}$ in an augmented space, which lifts the points into an additional space for smoother learning. Therefore, the dimensions M of $\mathbf{W}$ is an important hyperparameter, as it controls the output shape of the spatio-temporal representation learning that should be preserved for prediction. Figure 7 plots the training process of DyLand by varying the values of M from 0 to 5 – note that $M = 0$ means DyLand only model the dynamics of $\mathbf{S}$ for prediction. Increasing M can explore higher additional space of $\mathbf{W}$ to capture complex dependencies and mappings while preserving the topology of $\mathbf{W}$, allowing for lower losses and fewer function evaluations. However, M is restricted by the dimensions of $\mathbf{S}$, and become unstable for a larger value (e.g., 5) caused by increasing the complexity of functions required to be approximated in ODEs.

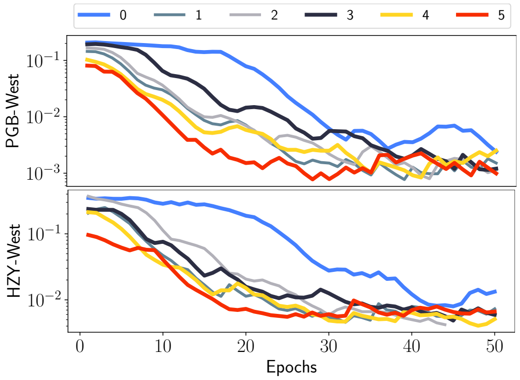


Figure 7: Training of DyLand in the first 50 epochs.

	Model	RMSE	MAE	ACC	$\mathbf{R}^2$	EVS
PBG-W	DyLand-$\mathbf{S}$	0.052	0.044	0.584	0.283	0.291
	DyLand-$\mathbf{W}$	0.019	0.015	0.983	0.497	0.500
	DyLand	0.013	0.010	0.993	0.538	0.540
HZY-W	DyLand-$\mathbf{S}$	0.076	0.057	0.527	0.407	0.458
	DyLand-$\mathbf{W}$	0.064	0.045	0.684	0.482	0.464
	DyLand	0.052	0.035	0.742	0.492	0.492

Table 3: Ablation results of DyLand on W and S co-training.

Conclusion

We presented DyLand, a general manifold learning model with application to forecasting land deformations, addressing the fundamental limitations of existing approaches in learning the spatio-temporal characteristics of InSAR point cloud data. Our novel dynamic manifold learning model for deformation distribution on the manifold surface with tractable density estimation, enables preserving both topology structure and deformation sensitivity for highly accurate land displacement prediction. DyLand also generalizes the dynamic process of learning deformation and representation through solving the ODEs of the co-evolution dynamic system. Extensive evaluations demonstrated the superior performance of DyLand over existing spatio-temporal GNN and static manifold models. In our future work, we will attempt to extend DyLand by incorporating meteorological data.

Acknowledgements

This work was supported by National Natural Science Foundation of China (Grant No. 62072077 and No. 62176043), and National Science Foundation SWIFT (Grant No. 2030249).

References

Brehmer, J.; and Cranmer, K. 2020. Flows for simultaneous manifold learning and density estimation. In *NeurIPS*, 442–453.

Carlà, T.; Intrieri, E.; Raspini, F.; Bardi, F.; Farina, P.; Ferretti, A.; Colombo, D.; Novali, F.; and Casagli, N. 2019. Perspectives on the prediction of catastrophic slope failures from satellite InSAR. *Scientific reports*, 9(1): 1–9.

Chen, C.; Li, C.; Chen, L.; Wang, W.; Pu, Y.; and Duke, L. C. 2018a. Continuous-Time Flows for Efficient Inference and Density Estimation. In *ICML*, 824–833.

Chen, R. T. Q.; Rubanova, Y.; Bettencourt, J.; and Duvenaud, D. K. 2018b. Neural Ordinary Differential Equations. In *NeurIPS*.

Chen, W.; Xie, X.; Peng, J.; Wang, J.; Duan, Z.; and Hong, H. 2017. GIS-based landslide susceptibility modelling: a comparative assessment of kernel logistic regression, Naïve-Bayes tree, and alternating decision tree models. *Geomatics, Natural Hazards and Risk*, 8(2): 950–973.

Cho, K.; van Merrienboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. arXiv:1406.1078.

Dai, F.; Lee, C.; and Ngai, Y. Y. 2002. Landslide risk assessment and management: an overview. *Engineering geology*, 64(1): 65–87.

de Blasio, F. V. 2020. *Introduction to Physics of Landslides*. Springer Nature.

Dong, J.; Zhang, L.; Liao, M.; and Gong, J. 2019. Improved correction of seasonal tropospheric delay in InSAR observations for landslide deformation monitoring. *Remote Sensing of Environment*, 233: 111370.

Donoho, D. L.; and Grimes, C. 2003. Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. *PNAS*, 100(10): 5591–5596.

Dormand, J. R.; and Prince, P. J. 1980. A family of embedded Runge-Kutta formulae. *Journal of computational and applied mathematics*, 6(1): 19–26.

Dupont, E.; Doucet, A.; and Teh, Y. W. 2019. Augmented Neural ODEs. In *NeurIPS*, 3140–3150.

Gan, B.-R.; Yang, X.-G.; and Zhou, J.-W. 2019. GIS-based remote sensing analysis of the spatial-temporal evolution of landslides in a hydropower reservoir in southwest China. *Geomatics, Natural Hazards and Risk*, 10(1): 2291–2312.

Gemici, M. C.; Rezende, D.; and Mohamed, S. 2016. Normalizing Flows on Riemannian Manifolds. arXiv:1611.02304.

Grathwohl, W.; Chen, R. T.; Bettencourt, J.; Sutskever, I.; and Duvenaud, D. 2018. FFJORD: Free-Form Continuous Dynamics for Scalable Reversible Generative Models. In *ICLR*.

Hajimoradlou, A.; Roberti, G.; and Poole, D. 2020. Predicting Landslides Using Locally Aligned Convolutional Neural Networks. In *IJCAI*, 3342–3348.

Hasanzadeh, A.; Hajiramezanali, E.; Narayanan, K.; Duffield, N.; Zhou, M.; and Qian, X. 2019. Semi-Implicit Graph Variational Auto-Encoders. In *NeurIPS*.

Hong, H.; Pradhan, B.; Jebur, M. N.; Bui, D. T.; Xu, C.; and Akgun, A. 2016. Spatial prediction of landslide hazard at the Luxi area (China) using support vector machines. *Environmental Earth Science*, 75(1): 40.

Hutchinson, M. F. 1989. A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines. *Communications in Statistics-Simulation and Computation*, 18(3): 1059–1076.

Kalantar, B.; Pradhan, B.; Naghibi, S. A.; Motevalli, A.; and Mansor, S. 2018. Assessment of the effects of training data selection on the landslide susceptibility mapping: a comparison between support vector machine (SVM), logistic regression (LR) and artificial neural networks (ANN). *Geomatics, Natural Hazards and Risk*, 9(1): 49–69.

Kayastha, P.; Dhital, M. R.; and De Smedt, F. 2013. Application of the analytical hierarchy process (AHP) for landslide susceptibility mapping: A case study from the Tinau watershed, west Nepal. *Computers & Geosciences*, 52: 398–408.

Kim, H.; Lee, H.; Kang, W. H.; Lee, J. Y.; and Kim, N. S. 2020. SoftFlow: Probabilistic Framework for Normalizing Flow on Manifolds. In *NeurIPS*, 16388–16397.

Kingma, D. P.; and Ba, J. 2017. Adam: A Method for Stochastic Optimization. arXiv:1412.6980.

Kipf, T. N.; and Welling, M. 2016. Variational Graph Auto-Encoders. arXiv:1611.07308.

Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *ICLR*.

Lei, T.; Zhang, Y.; Lv, Z.; Li, S.; Liu, S.; and Nandi, A. K. 2019. Landslide inventory mapping from bitemporal images using deep convolutional neural networks. *IEEE Geoscience and Remote Sensing Letters*, 16(6): 982–986.

Li, Y.; Yu, R.; Shahabi, C.; and Liu, Y. 2018. Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. In *ICLR*.

Maaten, L. v. d.; and Hinton, G. 2008. Visualizing data using t-SNE. *JMLR*, 9(Nov): 2579–2605.

McInnes, L.; Healy, J.; and Melville, J. 2020. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv:1802.03426.

Papamakarios, G.; Nalisnick, E.; Rezende, D. J.; Mohamed, S.; and Lakshminarayanan, B. 2021. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57): 1–64.

Rezende, D. J.; Papamakarios, G.; Racaniere, S.; Albergo, M.; Kanwar, G.; Shanahan, P.; and Cranmer, K. 2020. Normalizing Flows on Tori and Spheres. In *ICML*, 8083–8092.

Roweis, S. T.; and Saul, L. K. 2000. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500): 2323–2326.

Shi, W.; and Rajkumar, R. 2020. Point-GNN: Graph Neural Network for 3D Object Detection in a Point Cloud. In *CVPR*, 1711–1719.

Shirzadi, A.; Bui, D. T.; Pham, B. T.; Solaimani, K.; Chapi, K.; Kavian, A.; Shahabi, H.; and Revhaug, I. 2017. Shallow landslide susceptibility assessment using a novel hybrid intelligence approach. *Environmental Earth Science*, 76(2): 60.

Tenenbaum, J. B.; De Silva, V.; and Langford, J. C. 2000. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500): 2319–2323.

Thiery, Y.; Maquaire, O.; and Fressard, M. 2014. Application of expert rules in indirect approaches for landslide susceptibility assessment. *Landslides*, 11(3): 411–424.

Wan, S.; Lei, T.-C.; and Chou, T.-Y. 2012. A landslide expert system: image classification through integration of data mining approaches for multi-category analysis. *IJGIS*, 26(4): 747–770.

Wang, X.; Ma, Y.; Wang, Y.; Jin, W.; Wang, X.; Tang, J.; Jia, C.; and Yu, J. 2020. Traffic Flow Prediction via Spatial Temporal Graph Neural Network. In *WWW*, 1082–1092.

Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S. E.; Bronstein, M. M.; and Solomon, J. M. 2019. Dynamic graph cnn for learning on point clouds. *ACM TOG*, 38(5): 1–12.

Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; and Philip, S. Y. 2020. A comprehensive survey on graph neural networks. *TNNLS*, 32(1): 4–24.

Zhang, Z.; and Wang, J. 2007. MLLE: Modified Locally Linear Embedding Using Multiple Weights. In *NeurIPS*, 1593–1600.

Zhang, Z.; and Zha, H. 2004. Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. *SIAM journal on scientific computing*, 26(1): 313–338.

Zhao, L.; and Akoglu, L. 2020. PairNorm: Tackling Oversmoothing in GNNs. In *ICLR*.

Zhou, F.; Li, R.; Zhang, K.; and Trajcevski, G. 2021. Land Deformation Prediction via Slope-Aware Graph Neural Networks. In *AAAI*, 15033–15040.

Zhu, A.-X.; Wang, R.; Qiao, J.; Qin, C.-Z.; Chen, Y.; Liu, J.; Du, F.; Lin, Y.; and Zhu, T. 2014. An expert knowledge-based approach to landslide susceptibility mapping using GIS and fuzzy logic. *Geomorphology*, 214: 128–138.