DEEP DIFFUSION PROCESSES FOR ACTIVE LEARNING OF HYPERSPECTRAL IMAGES

Abiy Tasissa[1]*, Duc Nguyen*[2]*, James M. Murphy*[1]

[1]Department of Mathematics, Tufts University, USA
[2]Department of Mathematics, University of Maryland, College Park, USA

ABSTRACT

A method for active learning of hyperspectral images (HSI) is proposed, which combines deep learning with diffusion processes on graphs. A deep variational autoencoder extracts smoothed, denoised features from a high-dimensional HSI, which are then used to make labeling queries based on graph diffusion processes. The proposed method combines the robust representations of deep learning with the mathematical tractability of diffusion geometry, and leads to strong performance on real HSI.

***Index Terms*—** hyperspectral images, variational autoencoders, deep clustering, active learning, semisupervised learning, diffusion geometry

1. INTRODUCTION

Machine learning has provided revolutionary new tools for remote sensing, but state-of-the-art methods often require huge labeled training sets. In particular, supervised deep learning methods can achieve near-perfect labeling accuracy on high-dimensional hyperspectral images (HSI), provided large libraries of labeled pixels are available [1]. This hinders the practicality of these methods, as in many settings, data is collected at a pace that far exceeds human ability to generate corresponding labeled training data.

In order to account for this, methods that require only a very small number of labels are needed. The *active learning* regime is particularly attractive for HSI labeling problems. In active learning, an algorithm is provided with an unlabeled dataset, and the algorithm iteratively queries points for labels. By choosing query points intelligently, the active learning algorithm can yield the classification performance of a much larger training set chosen uniformly at random.

We propose an active learning method for HSI based on deep feature extraction and random walks on graphs. First, an unsupervised variational autoencoder is used to nonlinearly denoise and compress the high-dimensional HSI. Then, the resulting features are considered as vertices of a graph, and a Markov diffusion process on the graph is used to determine label queries and label all data points. The proposed method combines the efficient feature learning of deep autoencoders with the mathematical interpretability of graph diffusion processes, and leads to strong empirical performance on real HSI.

This research is partially supported by the US National Science Foundation grants NSF-DMS 1912737, NSF-DMS 1924513, and NSF-CCF 1934553.

2. BACKGROUND

2.1. Variational Autoencoders

In an autoencoder architecture, input data is cascaded through nonlinear layers to obtain a latent representation. The latent representation is then cascaded through nonlinear layers to obtain output data. These two stages respectively define the encoder and decoder. Typically, a loss function that enforces the reconstructed output to be similar to the input is minimized and the trained autoencoder learns a low-dimensional latent feature useful for downstream tasks. In contrast to the autoencoder, in the variational autoencoder (VAE) [2], the output of the encoder is not a deterministic map but parameters of a distribution. In particular, an encoding network maps $x \in \mathbb{R}^N$ and obtains parameters of the latent variable distribution $q(z|x)$. A latent feature z sampled from this distribution is an input to a decoder that outputs $\hat{x} \sim p(x|z)$. A typical prior for the distribution of the latent variable is a Gaussian random variable $p(z) \sim N(0, I)$. Given this, the VAE optimization consists of two terms: (i) a reconstruction loss $\mathcal{L}_1 = \mathbb{E}_{z\sim q(z|x)} \log(p(x|z))$ that enforces that the reconstructed output $\hat{x}$ is similar to the input x; and (ii) a Kullback-Leibler divergence loss $\mathcal{L}_2 = KL(q(z|x), N(0, I))$ that enforces that $q(z|x)$ agrees with $p(z)$. The encoder and decoder are jointly trained by maximizing the total loss $\mathcal{L}_1 + \mathcal{L}_2$.

2.2. Learning by Active Nonlinear Diffusion

The active learning algorithm employed in this paper is based on the ideas in [3, 4, 5]. In [6], the authors propose a semisupervised algorithm, learning by active nonlinear diffusion (LAND), that obtains the most important data points to query for labels. Important features of LAND are (i) it is a principled algorithm with provable performance guarantees; (ii) it accounts for nonlinear clusters possibly in high dimensions; and (iii) it is robust to noise and outliers [6].

We represent an HSI as $X = \{x_i\}_{i=1}^n \subset \mathbb{R}^N$ where each

pixel is a point in $\mathbb{R}^N$ where N is the number of spectral bands. Let $NN_k(x_i)$ denote the set of k-nearest neighbors of x_i in X using the Euclidean distance metric. The $n \times n$ weight matrix W is defined as $W_{ij} = \exp(-\|x_i - x_j\|_2^2/\sigma^2), x_j \in NN_k(x_i)$ with σ denoting a scale parameter. With this, the notion of the degree of x_i naturally follows as $\deg(x_i) := \sum_{x_j \in X} W_{ij}$. To define a random walk on X, we employ the $n \times n$ transition matrix $P_{ij} = W_{ij}/\deg(x_i)$. It can be easily verified that P has a spectral decomposition $\{(\lambda_\ell, \Psi_\ell)\}_{\ell=1}^n$. The *diffusion distance at time* t between $x_i, x_j \in X$ is defined as $D_t(x_i, x_j) = \sqrt{\sum_{\ell=1}^n \lambda_\ell^{2t}(\Psi_\ell(x_i) - \Psi_\ell(x_j))^2}$. We note that t tells us how long the diffusion process runs. In this paper, we use $t = 30$ for experiments.

The main part of the LAND algorithm is to identify points to query for labels. LAND uses a kernel density estimator (KDE) and diffusion geometry for this task. In particular, the KDE is defined as $p(x) = \sum_{y \in NN_k(x)} \exp(-\|x - y\|_2^2/\sigma_0^2)$ with σ_0 denoting a scale parameter. For $x \in X$, let

$$\rho_t(x) = \begin{cases} \min\limits_{p(y) \geq p(x), x \neq y} D_t(x, y), & x \neq \operatorname{argmax}\limits_z p(z), \\ \max\limits_{y \in X} D_t(x, y), & x = \operatorname{argmax}\limits_z p(z), \end{cases} \quad (1)$$

be the diffusion distance to the nearest neighbor of higher density. The maximizers of $\mathcal{D}_t(x) = p(x)\rho_t(x)$ are queried for labels. These labels are propagated to other data points by proceeding from high to low density and assigning each unlabeled point the same label as its D_t-nearest neighbor of higher density that is labeled; see Algorithm 1 and [6] for details.

2.3. Related Work

In recent years, deep generative methods, such as *generative adversarial network (GANs)* and *variational autoencoder networks (VAEs)*, have been used for feature extraction in many machine learning tasks [7, 8]. In the context of clustering, a set of methods, known as deep clustering, propose learning features of the data and clustering simultaneously, showing strong empirical results [9, 10, 11]. For HSI images, several works have employed different deep learning architectures to extract essential features for downstream tasks such as classification [12, 13, 14, 15, 16].

Active learning is a learning paradigm where the user has the ability to select the training data [17, 18]. The underlying idea is that a few informative training samples could be sufficient for training an algorithm and obtaining accurate results. This framework has been used in remote sensing for HSI image classification [19, 20, 21, 22]. The main idea in this paper is that the active learning process depends on the representation and geometry of the data. The closest work to ours is [23] where the authors combine active learning with VAEs. Therein, K-means clustering is first used to partition the space and then labels are acquired using uniform random sampling in each partition. Given the labels, a classifier is then trained in the latent space for the prediction task. One of the highlights of the proposed method is that the clustering algorithm LAND handles a broader class of cluster geometries than K-means does.

We note that in contrast to the similar work [24], our method is in the active learning framework and the feature extraction and diffusion process via LAND are decoupled.

3. PROPOSED ALGORITHM

We propose an active learning algorithm, VAE-LAND (see Algorithm 1), which has two main stages. The first stage is feature extraction of an unlabeled high-dimensional dataset using a VAE. The second stage employs the LAND algorithm to infer the true labels. The proposed algorithm combines the power of VAEs to extract features with diffusion geometry on graphs to find impactful labels to query, which then propagate to other points.

Algorithm 1: Variational Autoencoder Learning by Active Nonlinear Diffusion (VAE-LAND)

Input: $\{x_i\}_{i=1}^n$ (Unlabeled Data); t (Time Parameter); B (Budget); $\mathcal{O}$ (Labeling Oracle)

Output: Y (Labels)

Run VAE on unlabeled data to obtain the latent representation $\{\hat{x}_i\}_{i=1}^n$.
Compute P and $\{(\lambda_\ell, \psi_\ell)\}_{\ell=1}^M$ using $\{\hat{x}_i\}_{i=1}^n$.
Compute kernel density estimate $\{p(\hat{x}_i)\}_{i=1}^n$ and $\{\rho_t(\hat{x}_i)\}_{i=1}^n$ (1);
Compute $\mathcal{D}_t(\hat{x}_i) = p(\hat{x}_i)\rho_t(\hat{x}_i)$.
Sort the data in decreasing $\mathcal{D}_t$ value to acquire the ordering $\{\hat{x}_{m_i}\}_{i=1}^n$.
for $i = 1 : B$ **do**
    Query $\mathcal{O}$ for the label $L(\hat{x}_{m_i})$ of $\hat{x}_{m_i}$.
    Set $Y(\hat{x}_{m_i}) = L(\hat{x}_{m_i})$.
end for
Sort X according to p in decreasing order as $\{\hat{x}_{\ell_i}\}_{i=1}^n$.
for $i = 1 : n$ **do**
    if $Y(\hat{x}_{\ell_i}) = 0$ **then**
        $Y(\hat{x}_{\ell_i}) = Y(z_i)$, $z_i = \operatorname{argmin}_z\{D_t(z, \hat{x}_{\ell_i}) \mid p(z) > p(\hat{x}_{\ell_i})$ and $Y(z) > 0\}$.
    end if
end for

4. EXPERIMENTAL RESULTS

We demonstrate the accuracy of the proposed algorithm experimentally. The training of the VAE is done using Tensorflow in Python. For doing the active learning via LAND, we use the publicly available MATLAB code at

https://jmurphy.math.tufts.edu/Code/. Our code can be found at https://github.com/abiy-tasissa/VAE-LAND. Our test HSI dataset is the Salinas A hyperspectral dataset. The Salinas scene was captured over Salinas Valley, California. The image has a spatial resolution of 3.7-meter pixels and contains 224 spectral bands. The ground truth consists of 16 classes. We consider the Salinas A dataset, which is a subset of the Salinas dataset, and contains 6 classes. The Salinas A dataset and the ground truth data are publicly available (http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes#Salinas-A_scene). Figure 1 shows a visual of the high-dimensional data and the ground truth labels. The performance of the algorithm is assessed using overall accuracy, defined as the ratio of correctly estimated labels to total number of labels after optimally aligning with the ground truth. The Salinas A HSI dataset

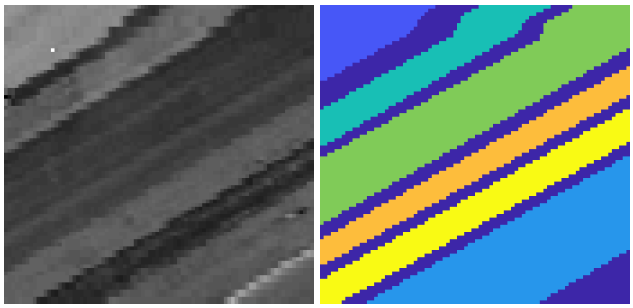

Fig. 1: The 86×83 Salinas A HSI data consists of 6 classes. *Left:* the sum of all spectral bands. *Right:* the ground truth.

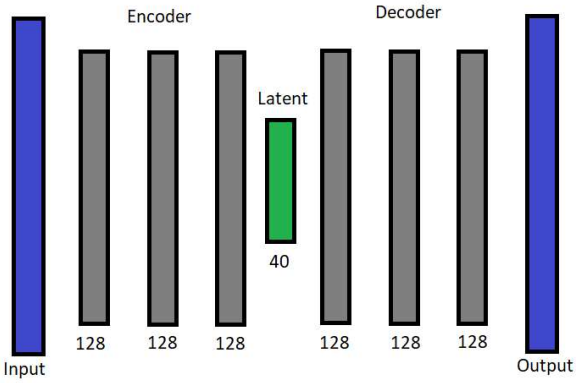

Fig. 2: A schematic of the VAE architecture. Input is a vector in $\mathbb{R}^{224}$. The encoder and decoder are fully connected neural networks. Both have three layers with 128 neurons in each layer. For all layers, the activation function is the rectified linear unit (ReLU). The input is cascaded through an encoder. The extracted latent feature in $\mathbb{R}^{40}$ is then cascaded through the decoder to obtain the output vector in $\mathbb{R}^{224}$. Unlike the standard autoencoder, the extraction of the latent feature is not deterministic (see Section 2.1 for discussion.)

of size $83 \times 86 \times 224$ is represented as a point cloud of size 7138×224. We use the unlabeled data to learn a latent space representation of Salinas A in $\mathbb{R}^{40}$ dimensions. A schematic of the VAE architecture is shown in Figure 2. We optimize the VAE loss function using the Adam algorithm with learning rate set to 10^{-4}. After training the VAE, we input the optimal latent space representation of the Salinas A dataset to the LAND algorithm for the task of inferring the ground truth labels of the HSI data. We compare our result to the standard LAND algorithm that labels the Salinas A dataset in its original representation. Since LAND is an active learning framework, we consider varying number of labeled data points ranging from 10 to 2000. In addition, we compare the active learning methods to query the samples with randomly selected training data. Figure 1 compares the performance of LAND and performance of VAE-LAND. First, for both VAE-LAND and standard LAND, LAND queries lead to significantly better accuracy than random queries. The proposed algorithm, VAE-LAND, attains an accuracy of 96.97% with just 10 labeled points. This is a 12.5% improvement to the accuracy of the standard LAND algorithm for the same number of labeled points. The standard LAND algorithm requires 400 labeled points to reach accuracy of 90% while for the same number of labeled points, VAE-LAND has an accuracy of 98.35%.

Complexity and run time: The complexity of LAND is $O(C_{NN} + nK_{NN} + n\log(n))$ where C_{NN} is the cost of computing all K_{NN} nearest neighbours [6]. The computational cost of VAE is difficult to estimate as it depends on several factors (e.g architecture, activation function, choice of SGD algorithm). In our numerical experiments, the cost of VAE is the dominating cost. Since LAND runs on low-dimensional features extracted from VAE, it is efficient.

5. CONCLUSIONS AND FUTURE DIRECTIONS

The proposed active learning algorithm, VAE-LAND, improves over the standard LAND and gives accurate results even when the number of queries are limited. The method uses VAE to generate good features, and uses the diffusion geometry-based LAND algorithm to determine query points. The LAND algorithm then uses these queried labels to predict the labels of the unlabeled data samples. In future work, we shall explore data models for which the algorithm has theoretical performance guarantees.

6. REFERENCES

[1] X.X. Zhu, D. Tuia, L. Mou, G.-S. Xia, L. Zhang, F. Xu, and F. Fraundorfer, "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, 2017.

[2] D.P. Kingma and M. Welling, "Auto-encoding variational Bayes," *Stat*, vol. 1050, 2014.

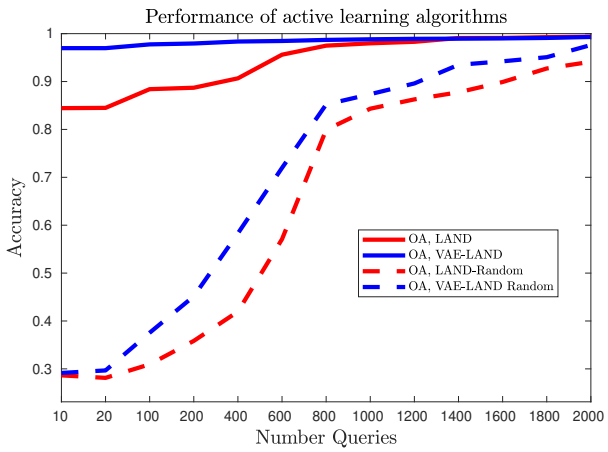

Fig. 3: For the Salinas A dataset, the performance of VAE-LAND learning achieves a higher accuracy than the standard LAND algorithm. With just 10 points, the overall accuracy of VAE-LAND is 96.97%, a 12.5% improvement to the competitive LAND algorithm. Both VAE-LAND and LAND obtain significantly better results than using randomly selected training instances.

[3] J.M. Murphy and M. Maggioni, "Unsupervised clustering and active learning of hyperspectral images with nonlinear diffusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 3, 2019.

[4] M. Maggioni and J.M. Murphy, "Learning by unsupervised nonlinear diffusion," *J. Mach. Learn. Res.*, vol. 20, no. 160, 2019.

[5] J.M. Murphy and M. Maggioni, "Spectral-spatial diffusion geometry for hyperspectral image clustering," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 7, 2020.

[6] M. Maggioni and J.M. Murphy, "Learning by active nonlinear diffusion," *Foundations of Data Sci.*, vol. 1, no. 3, 2019.

[7] M.E. Abbasnejad, A. Dick, and A. van den Hengel, "Infinite variational autoencoder for semi-supervised learning," in *CVPR*, 2017.

[8] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey, "Adversarial autoencoders," *arXiv preprint arXiv:1511.05644*, 2015.

[9] F. Tian, B. Gao, Q. Cui, E. Chen, and T.-Y. Liu, "Learning deep representations for graph clustering," in *AAAI*, 2014.

[10] C. Song, F. Liu, Y. Huang, L. Wang, and T. Tan, "Auto-encoder based data clustering," in *Iberoamerican Congress Pattern Recognit.* Springer, 2013.

[11] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *ICML*, 2016.

[12] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, 2014.

[13] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, 2016.

[14] Y. Li, H. Zhang, and Q. Shen, "Spectral–spatial classification of hyperspectral imagery with 3D convolutional neural network," *Remote Sens.*, vol. 9, no. 1, 2017.

[15] M. He, B. Li, and H. Chen, "Multi-scale 3D deep convolutional neural network for hyperspectral image classification," in *ICIP*. IEEE, 2017.

[16] M.E. Paoletti, J..M. Haut, J. Plaza, and A. Plaza, "Deep learning classifiers for hyperspectral imaging: A review," *ISPRS J. Photogram. Remote Sens.*, vol. 158, 2019.

[17] D.A. Cohn, Z. Ghahramani, and M.I. Jordan, "Active learning with statistical models," in *NIPS*, 1995.

[18] D. MacKay, "Information-based objective functions for active data selection," *Neural Comput.*, vol. 4, no. 4, 1992.

[19] P. Liu, H. Zhang, and K.B. Eom, "Active deep learning for classification of hyperspectral images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 2, 2016.

[20] Z. Wang, B. Du, L. Zhang, L. Zhang, and X. Jia, "A novel semisupervised active-learning algorithm for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 55, no. 6, 2017.

[21] J.M. Murphy and M. Maggioni, "Iterative active learning with diffusion geometry for hyperspectral images," in *WHISPERS*. 2018, IEEE.

[22] D. Tuia, F. Ratle, F. Pacifici, M.F. Kanevski, and W.J. Emery, "Active learning methods for remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 7, 2009.

[23] F. Pourkamali-Anaraki and M.B. Wakin, "The effectiveness of variational autoencoders for active learning," *arXiv preprint arXiv:1911.07716*, 2019.

[24] H. Li, O. Lindenbaum, X. Cheng, and A. Cloninger, "Variational diffusion autoencoders with random walk sampling," in *ECCV*. Springer, 2020.