DEEP DIFFUSION PROCESSES FOR ACTIVE LEARNING OF HYPERSPECTRAL IMAGES

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

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

2. VARIATIONAL AUTOENCODERS

We now describe VAEs following the exposition in [2]. Let $x \in \mathbb{R}^N$ and consider a latent variable model $p_\theta(x) = \int p(x|z; \theta)p(z)\, dz$. The variable $z \in \mathbb{R}^D$, $D \ll N$, is a latent variable and provides a low-dimensional parametrization of the data x. The goal of VAE is to maximize the probability of data samples generated as $p(x)$. In the inference part of VAE, the training procedure maps a data point x to its latent representation z such that the z adheres to the distribution $p(z)$.

Since the posterior distribution $p(z|x)$ and $p(z)$ are unknown, VAE makes the assumption that $p(z|x) = q(z|x, \phi)$ where ϕ are parameters to be learned from the network. To enforce that the variational distribution $q(z|x, \phi)$ agrees with $p(z)$, the optimization $\min_\phi KL(q(z|x, \phi), p(z))$ is considered, where KL is the Kullback-Leibler divergence. With this, $q(z|x)$ can be interpreted as the encoder part of VAE providing us a latent representation z given a sample x.

The generative part of VAE considers the reconstruction of a data sample given the latent representation z. In particular, the goal of the training procedure is to map a latent variable z to a data sample $\hat{x}$ such that $\hat{x}$ is similar to the true data sample x. This naturally motivates the maximization of $p(x|z, \theta)$ with θ denoting the network parameters. Formally, we consider the optimization program $\max_\theta \mathbb{E}_{z \sim q(z|x,\phi)} \log p(x|z; \theta)$. With this, $p(x|z)$ can be interpreted as the decoder part of VAE. The inference and generative parts of VAE i.e the encoder and decoder model can be jointly trained by maximizing the loss function

$$\mathcal{L}(x, z; \theta, \phi) = \mathbb{E}_{z \sim q(z|x,\phi)} \log p(x|z; \theta) - KL(q(z|x, \phi), p(z))$$

In summary, the optimal network parameters θ^* and ϕ^* are obtained as $(\theta^*, \phi^*) = \text{argmax}_{\theta, \phi} \mathcal{L}(x, z; \theta, \phi)$, optimized using stochastic gradient descent. Intuitively, the encoding of $x \in \mathbb{R}^N$ by $z \in \mathbb{R}^D$ for $D \ll N$ is an efficient low-dimensional representation of the data which ideally captures

the underlying statistics and geometry of the data. In our proposed methodology, the latent space representation resulting from a trained VAE is used as input to a clustering algorithm which suggests candidate points to query for labels.

2.1. Learning by Active Nonlinear Diffusion

In this paper, we use an active learning algorithm for high dimensional data based on diffusion geometry [3, 4, 5]. The algorithm, learning by active nonlinear diffusion (LAND) [6], is a semisupervised algorithm that takes into account the geometry of the data to identify the most important points to be query for labels. A key advantage of LAND is that it provides rigorous theoretical performance guarantees and handles general clusters that could be nonlinear, live in a high ambient dimension, and may be corrupted by noise and outliers [6].

Consider an HSI $X = \{x_i\}_{i=1}^n \subset \mathbb{R}^m$, where each pixel is represented as a point in $\mathbb{R}^m$ where m is the number of spectral bands. Let W to be the $n \times n$ weight matrix defined as $W_{ij} = \exp(-\|x_i - x_j\|_2^2/\sigma^2), x_j \in NN_k(x_i)$ and $W(x_i, x_j) = 0$ otherwise, where $NN_k(x_i)$ is the set of k-nearest neighbors of x_i in X with respect to Euclidean distance and σ is a scale parameter. Given the weight matrix, the degree of x is $\deg(x_i) := \sum_{x_j \in X} W_{ij}$. A random walk on X can be defined using the $n \times n$ transition matrix $P_{ij} = W_{ij}/\deg(x_i)$. By construction, P has a spectral decomposition $\{(\lambda_i, \Psi_i)\}_{ni=1}^n$, and we define the *diffusion distance at time t* between $x_i, x_j \in X$ as $D_t(x_i, x_j) = \sqrt{\sum_{\ell=1}^n \lambda_\ell^{2t}(\Psi_\ell(x_i) - \Psi_\ell(x_j))^2}$. The parameter t in diffusion distance depends on a parameter t informs how long the diffusion process runs. In this paper, we set t to be 30; diffusion learning is relatively robust to choice of t [3].

An important part of the LAND algorithm is to determine the data points which one should query for labels. This is achieved using a kernel density estimator and diffusion geometry. Consider the density estimator $p(x) = \sum_{y \in NN_k(x)} \exp(-\|x - y\|_2^2/\sigma_0^2)$ with σ_0 denoting a scale parameter. Let

$$\rho_t(x) = \begin{cases} \min_{p(y) \geq p(x), x \neq y} D_t(x, y), & x \neq \operatorname*{argmax}_z p(z) \\ \max_{y \in X} D_t(x, y), & x = \operatorname*{argmax}_z p(z) \end{cases} \quad (1)$$

be the (t-dependent) diffusion distance between a point and its nearest diffusion neighbor of higher density if x is not the maximizer of $p(x)$, and the maximum diffusion distance to another point if x is the maximizer of $p(x)$. The modes of the data are determined as the maximizers of $\mathcal{D}_t(x) = p(x)\rho_t(x)$. A large $\mathcal{D}_t$ value indicates a high-density point that is D_t-far from other high density points. In LAND, the modes of X are characterized as maximizers of $\mathcal{D}_t$. Diffusion distances and density can also employed to label all other points relative to these modes. We refer the interested reader to [6].

2.2. Related Work

Extracting essential features of data is an important part of both supervised and unsupervised machine learning problems. For example, the unsupervised technique of Gaussian mixture models (GMM) assumes that the data under consideration is an i.i.d. sample from a mixture of Gaussians, and the parameters of the GMM are estimated by expectation maximization [7]. In recent years, deep generative methods, such as *generative adversarial network (GANs)* and *variational autoencoder networks (VAEs)*, have been used as feature extraction tools and successfully applied to many machine learning tasks [8, 9]. In the context of the clustering problem, a set of methods, known as deep clustering, propose learning features of the data and clustering simultaneously, showing strong empirical results [10, 11, 12, 13]. In the context of HSI images, several works have employed different deep learning architectures to extract essential features for downstream tasks such as classification [14, 15, 16, 17, 18, 19].

Active learning is a learning paradigm where the user has the ability to select the training data[20, 21]. Consider unlabeled data $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$ in $\mathbb{R}^m$ with corresponding true labels $\{y_1, y_2, ..., y_n\}$. In active learning, the user can obtain labels for at most $B < n$ of the data points. Hereafter, we refer to B as the fixed budget the user has access to. The underlying idea is that a few informative training samples could be sufficient for training an algorithm and obtaining accurate results. Given this, active learning methods differ in the way these informative samples are chosen. This learning framework has been used in remote sensing for hyperspectral image classification [22, 23, 24, 25]. This list is by no means comprehensive and we refer the interested reader to the survey in [26]. The main idea in this paper is that the active learning process depends on the representation and geometry of the data. For the former, we use deep neural networks to extract low dimensional features. To sample training data, restricted to the budget B, we employ a diffusion process based approach that is provably robust to geometry of the data. The closest work to ours is [27] where the authors combine active learning with variational autoencoders. 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. The underlying assumption is that the K-means partitions represent the structure of the latent space, and empirical results in [27] show the superiority of this approach to doing active learning via K-means partitions in the original space. Since the performance of the active learning algorithm hinges on how well the geometry of the latent space is represented, the choice of the clustering method is crucial. One of the highlights of the proposed method is that the clustering algorithm LAND is theoretically grounded and handles a more broad class of cluster geometries than K-means does.

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 variational autoencoder. The second stage employs the LAND active learning diffusion based clustering algorithm to infer the true labels. The second part of the algorithm concerns deploying clustering algorithms on the latent representation of the data. Since our algorithm is in the active learning framework, we first acquire labels for small subset of the latent data. Different active learning algorithms differ in the ways in which one determines the subset of the data to query for labels. The proposed algorithm combines the power of variational autoencoders to extract features and uses 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 (VALAND)

Input: $\{x_i\}_{i=1}^n$ (Unlabeled Data); $\{(\lambda_\ell, \psi_\ell)\}_{\ell=1}^M$ (Spectral Decomposition of P); $\{p(x_i)\}_{i=1}^n$ (Kernel Density Estimate); $\{\rho_t(x_i)\}_{i=1}^n$ (1); t (Time Parameter); B (Budget); $\mathcal{O}$ (Labeling Oracle)

Output: Y (Labels)

1: Run VAE on unlabeled data to obtain the latent representation $X = \{\hat{x}_i\}_{i=1}^n$.
2: Compute $\mathcal{D}_t(\hat{x}_i) = p(\hat{x}_i)\rho_t(\hat{x}_i)$.
3: Sort the data in decreasing $\mathcal{D}_t$ value to acquire the ordering $\{\hat{x}_{m_i}\}_{i=1}^n$.
4: **for** $i = 1 : B$ **do**
5:    Query $\mathcal{O}$ for the label $L(\hat{x}_{m_i})$ of $\hat{x}_{m_i}$.
6:    Set $Y(\hat{x}_{m_i}) = L(\hat{x}_{m_i})$.
7: **end for**
8: Sort X according to $p(\hat{x})$ in decreasing order as $\{\hat{x}_{\ell_i}\}_{i=1}^n$.
9: **for** $i = 1 : n$ **do**
10:    **if** $Y(\hat{x}_{\ell_i}) = 0$ **then**
11:        $Y(\hat{x}_{\ell_i}) = Y(z_i)$, $z_i = \underset{z}{\mathrm{argmin}}\{D_t(z, \hat{x}_{\ell_i}) \mid p(z) > p(\hat{x}_{\ell_i}) \text{ and } Y(z) > 0\}$.
12:    **end if**
13: **end for**

---

4. EXPERIMENTAL RESULTS

We demonstrate the accuracy of the proposed algorithm by running numerical experiments. The training of the VAE is done using Tensorflow in Python. For doing the clustering 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 2 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.
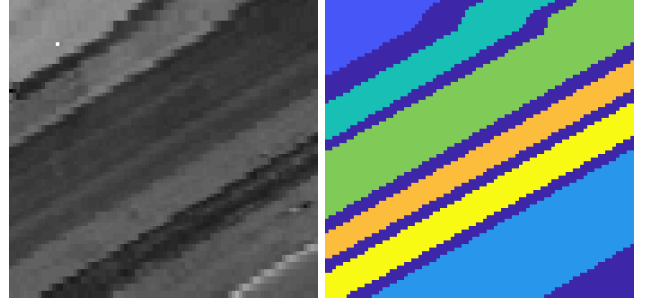
Fig. 1: The 86×83 Salinas A HSI data consists of 6 classes. Left: the sum of all spectral bands. Right: the ground truth.

Fig. 2: A schematics of the VAE archiecture. Input is cascaded through an encoder, three dense layers of size 128. The extraccted latent feature is fed to the decoder, three dense layers of size 128, to obtain the output. Note, unlike the standard autoencoder, the extraction of the latent feature is not deterministic.

The Salinas A HSI dataset 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. The encoder network consists of three dense layers with 128 units. The activation function is the rectified linear unit (RELU). The loss function is optimized using the Adam algorithm with learning rate set to 0.0001. After training the VAE, we input the 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 "plain" LAND algorithm that simply clusters 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 want to compare the active learning methods to query the samples with randomly selected training data. Figure 2 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,
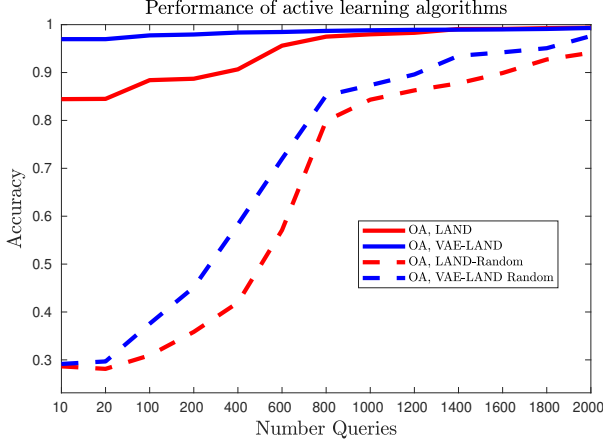
Fig. 3: For the Salinas A dataset, the performance of variational autoencoder LAND learning achieves a higher accuracy than the standard LAND algorithm. With just 10 points, the overall accuracy of VAE-LAND is 96.97%, a 12.5% improvement to the competitive LAND algorithm. Both VAE-LAND and LAND obtain significantly better results than using randomly selected training instances.

VAE-LAND, attains an accuracy of 96.97% with just 10 labeled points. This is a 12.5% improvement to the accuracy of the plain 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 this queried labels to predict the labels of the unlabeled data samples. In future work, we would like to explore the kind of 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 Geoscience and Remote Sensing Magazine*, vol. 5, no. 4, pp. 8–36, 2017.

[2] T. Yang, G. Arvanitidis, D. Fu, X. Li, and S. Hauberg, "Geodesic clustering in deep generative models," *arXiv preprint arXiv:1809.04747*, 2018.

[3] J.M. Murphy and M. Maggioni, "Unsupervised clustering and active learning of hyperspectral images with nonlinear diffusion," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 3, pp. 1829–1845, 2019.

[4] M. Maggioni and J.M. Murphy, "Learning by unsupervised nonlinear diffusion," *Journal of Machine Learning Research*, vol. 20, no. 160, pp. 1–56, 2019.

[5] J.M. Murphy and M. Maggioni, "Spectral-spatial diffusion geometry for hyperspectral image clustering," *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 7, pp. 1243–1247, 2020.

[6] M. Maggioni and J.M. Murphy, "Learning by active nonlinear diffusion," *Foundations of Data Science*, vol. 1, no. 3, pp. 271, 2019.

[7] K.P. Murphy, *Machine learning: a probabilistic perspective*, MIT press, 2012.

[8] M.E. Abbasnejad, A. Dick, and A. van den Hengel, "Infinite variational autoencoder for semi-supervised learning," in *CVPR*, 2017, pp. 5888–5897.

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

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

[11] C. Song, F. Liu, Y. Huang, L. Wang, and T. Tan, "Auto-encoder based data clustering," in *Iberoamerican Congress on Pattern Recognition*. Springer, 2013, pp. 117–124.

[12] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *ICML*, 2016, pp. 478–487.

[13] Z. Jiang, Y. Zheng, H. Tan, B. Tang, and H. Zhou, "Variational deep embedding: an unsupervised and generative approach to clustering," in *IJCAI*, 2017, pp. 1965–1972.

[14] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," *IEEE Journal of Selected topics in applied earth observations and remote sensing*, vol. 7, no. 6, pp. 2094–2107, 2014.

[15] 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 Transactions on Geoscience and Remote Sensing*, vol. 54, no. 10, pp. 6232–6251, 2016.

[16] Y. Li, H. Zhang, and Q. Shen, "Spectral–spatial classification of hyperspectral imagery with 3d convolutional neural network," *Remote Sensing*, vol. 9, no. 1, pp. 67, 2017.

[17] M. He, B. Li, and H. Chen, "Multi-scale 3d deep convolutional neural network for hyperspectral image classification," in *ICIP*. IEEE, 2017, pp. 3904–3908.

[18] M.E. Paoletti, J..M. Haut, J. Plaza, and A. Plaza, "Deep learning classifiers for hyperspectral imaging: A review," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 158, pp. 279–317, 2019.

[19] H. Zhang, L. Meng, X. Wei, X. Tang, X. Tang, X. Wang, B. Jin, and W. Yao, "1d-convolutional capsule network for hyperspectral image classification," *arXiv preprint arXiv:1903.09834*, 2019.

[20] D.A. Cohn, Z. Ghahramani, and M.I. Jordan, "Active learning with statistical models," in *NIPS*, 1995, pp. 705–712.

[21] D. MacKay, "Information-based objective functions for active data selection," *Neural Computation*, vol. 4, no. 4, pp. 590–604, 1992.

[22] P. Liu, H. Zhang, and K.B. Eom, "Active deep learning for classification of hyperspectral images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 2, pp. 712–724, 2016.

[23] Z. Wang, B. Du, L. Zhang, L. Zhang, and X. Jia, "A novel semisupervised active-learning algorithm for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 6, pp. 3071–3083, 2017.

[24] J.M. Murphy and M. Maggioni, "Iterative active learning with diffusion geometry for hyperspectral images," in *WHISPERS*, 2018, pp. 1–5.

[25] D. Tuia, F. Ratle, F. Pacifici, M.F. Kanevski, and W.J. Emery, "Active learning methods for remote sensing image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 7, pp. 2218–2232, 2009.

[26] D. Tuia, M. Volpi, L. Copa, M. Kanevski, and J. Munoz-Mari, "A survey of active learning algorithms for supervised remote sensing image classification," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 3, pp. 606–617, 2011.

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