COMPUTATIONAL METHODS IN MACHINE LEARNING: TRANSPORT MODEL, HAAR WAVELET, DNA CLASSIFICATION, AND MRI

Franck Olivier Ndjakou Njeunje

Applied Mathematics, Statistics, and Scientific Computation
Norbert Wiener Center
Department of Mathematics
University of Maryland, College Park

Outline

1. Overview
2. Introduction
3. Transport by advection
4. Experiments and results

Outline

1 Overview

2 Introduction

3 Transport by advection

4 Experiments and results

Overview

My dissertation includes the following topics:

- **Haar approximation from within for $L^p(\mathbb{R}^d)$, $0 < p < 1$**

Theorem (J. Benedetto and F. Njeunje)

Let $f \in L^p(\mathbb{R})$, where $0 < p < 1$ suppose f is a continuous function on $\mathbb{R}$, with $\operatorname{supp} f \subseteq [A, B]$. Then, for all $\epsilon > 0$, there is an $M = M(\epsilon)$, and there is a sequence of sums,

$$f_{M,k} = \sum_{(i,j)\in S_{M,k}} \tilde{a}_{i,j}\widetilde{\psi}_{i,j}, \quad \tilde{a}_{i,j} \in \mathbb{C},$$

indexed by $k \geq 1$, where $S_{M,k} \subseteq \mathbb{Z} \times \mathbb{Z}$ and $\operatorname{card} S_{M,k} < \infty$, with the following properties:

$$\text{if } (i,j) \in S_{M,k} \text{ then } \operatorname{supp} \widetilde{\psi}_{i,j} \subseteq \operatorname{supp} f,$$

and

$$\exists K = K(\epsilon) \text{ such that } \forall k > K, \ \|f - f_{M,k}\|_p < \epsilon.$$

Overview

My dissertation includes the following topics:

- **Classification of multiton enhancers**
  - In collaboration with Dr. Ivan Ovcharenko and his research group at the National Institutes of Health (NIH).
  - Enhancers are particular deoxyribonucleic acid (DNA) segments that increase or enhance the likelihood of gene expression.
  - Singletons vs. multitons.
  - We constructed a classifier using support vector machine to identify multitons having similar characteristics to singletons with high probability.
- **Analysis of T_2-store-T_2 magnetic resonance relaxometry with N exchanging sites**
  - In collaboration with Dr. Richard G. Spencer and his research group at NIH.
  - Magnetic resonance imaging (MRI) is a tool used for diagnosing anatomy and pathology, including osteoarthritis.
  - We successfully extended the analysis of the magnetization signal from 2 sites to N sites.
- **Transport operator on graph**
  - In collaboration with Prof. Wojciech Czaja and Prof. Pierre-Emmanuel Jabin.
  - This presentation contains material from this topic.

Outline

1 Overview

2 Introduction

3 Transport by advection

4 Experiments and results

Introduction

- The curse of dimensionality
  - This expression was coined by Richard Bellman and refers to the problem caused by the exponential increase in volume associated with adding extra dimensions to a mathematical space.
  - In data science this means that the number of observations needed to obtain favorable results grows exponentially with the number of dimensions.

- Dimension reduction (DR):
  - Principal component analysis (PCA), by Pearson[1].
    - Based on the covariance matrix.
    - Search of the orthogonal directions of greatest variance explaining as much of the data as possible.
  - Kernel PCA, by Schölkopf[2].
    - Non-linear adaptation of PCA.
    - A great number of non-linear DR algorithms are special cases of kernel PCA.

[1] K. Pearson, On lines and planes of closest fit to systems of point in space, Philosophical Magazine 2 (1901), no. 11, 559-572.

[2] B. Schölkopf, A. Smola, and K-R. Müller, Kernel principal component analysis, International Conference on Artificial Neural Networks, Springer, 1997, pp. 583-588.

Introduction

- Dimension reduction (DR):
  - Diffusion maps (DIF), by Coifman and Lafon[3].
    - Diffusion maps are constructed using eigenfunctions of Markov matrices.
    - They generate efficient representations of complex geometric structures.
  - Isomap (ISO), by Tenenbaum[4].
    - Based on the geodesic distance between points measured along the manifold.
  - Laplacian eigenmaps (LE), by Belkin and Niyogi[5].
    - Preserves local information embedded in low dimensional manifold.
  - Schroedinger eigenmaps (SE), by Czaja and Ehler[6].
    - Semi-supervised generalization of LE.
    - Uses barrier potential to stir the diffusion process.

[3] R. R. Coifman and S. Lafon, Diffusion maps, Applied and Computational Harmonic Analysis 21 (2006), no. 1, 5-30.

[4] J. B. Tenenbaum, V. De Silva, and J. C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (2000), no. 5500, 2319-2323.

[5] M. Belkin and P. Niyogi, Laplacian eigenmaps and spectral techniques for embedding and clustering, Advances in Neural Information Processing Systems, 2002, pp. 585-591.

[6] W. Czaja and M. Ehler, Schroedinger eigenmaps for the analysis of biomedical data, IEEE Transactions on Pattern Analysis and Machine Intelligence 35 (2013), no. 5, 1274-1280.

Laplacian eigenmaps: optimization problem

Given a set of n points $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ in $\mathbb{R}^d$, the goal is to find an optimal embedding for these points in a lower m-dimensional space where $m \ll d$, while preserving local information.

The embedding is given by the $n \times m$ matrix $\mathcal{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_m]$, where the i^{th} row corresponds to the embedded coordinates of the i^{th} points $\mathbf{x}_i$. The objective to the minimization problem[7] is written as

$$\sum_{i,j} \|\mathbf{y}^{(i)} - \mathbf{y}^{(j)}\|^2 w_{ij} = \mathrm{tr}\,(\mathcal{Y}^T L \mathcal{Y}), \tag{1}$$

where

- $\mathbf{y}^{(i)} = [\mathbf{y}_1(i), \ldots, \mathbf{y}_m(i)]^T$ is the m-dimensional representation of the i^{th} point $\mathbf{x}_i$.
- With the appropriate choice of weights w_{ij}, minimizing (1) ensures that adjacent points remain close together after the mapping.

[7] M. Belkin and P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Computation 15 (2003), no. 6, 1373-1396.

Laplacian eigenmaps: algorithm

The LE algorithm we will be using in our work involves the following steps:

- **Step 1:** Construct the adjacency graph using the k-nearest neighbor (kNN) algorithm. This is done by putting an edge connecting nodes i and j given that $\mathbf{x}_i$ is among the k nearest neighbors of $\mathbf{x}_j$.
- **Step 2:** Define a graph Laplacian, L , using the weight matrix, W . The weights in W are chosen using the heat kernel with parameter σ. If nodes i and j are connected,

$$w_{ij} = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right);$$

otherwise, $w_{ij} = 0$. The graph Laplacian is given by

$$L = D - W,$$

where D is a diagonal matrix with entries $d_{ii} = \sum_i w_{ij}$.

Laplacian eigenmaps: algorithm (continues)

- **Step 3:** Find the m-dimensional mapping by solving the generalized eigenvector problem,

$$L\mathbf{f} = \lambda D\mathbf{f}, \tag{2}$$

where $\mathbf{f}$ is a vector in $\mathbb{R}^n$ and λ is a real number. Let $\{\mathbf{f}_0, \mathbf{f}_1, \ldots, \mathbf{f}_{n-1}\}$ be the solution set to (2) written in ascending order according to their eigenvalues $\{\lambda_0, \lambda_1, \ldots, \lambda_{n-1}\}$. The m-dimensional Euclidean space mapping is given by

$$\mathbf{x}_i \rightarrow [\mathbf{f}_1(i), \mathbf{f}_2(i), \ldots, \mathbf{f}_m(i)].$$

Example: LE vs PCA

- Laplacian eigenmaps is able to represent the data in a reasonable manner preserving local information.
- PCA fails to capture the true nature of the data and simply project it to a 2-dimensional space.

Figure 1: The leftmost plot represents a set of 2000 3-dimensional points sitting on a swiss roll; the middle plot represents the embedding in 2-dimension using principal component analysis (PCA); and the rightmost plot represents the same embedding using Laplacian eigenmaps (LE) with $k = 12$ (number of neighbors per node) and $\sigma = 1$.

Schroedinger eigenmaps: optimization problem

Czaja and Ehler proposed the Schroedinger eigenmaps (SE) algorithm:

- Semi-supervised generalization to the LE algorithm using partial knowledge about the ground truth of the data set.
- The minimization problem[8]

$$\min_{\mathcal{Y}^T D \mathcal{Y} = \mathbf{I}} \frac{1}{2} \sum_{i,j} \|\mathbf{y}^{(i)} - \mathbf{y}^{(j)}\|^2 w_{ij} + \alpha \sum_i V(i) \|\mathbf{y}^{(i)}\|^2, \tag{3}$$

  where V is the diagonal matrix with entries $V(1)$ through $V(n)$.
- The second component of the sum (3) add an extra level of clustering on the representation $\mathbf{y}^{(i)}$ which are associated with large value of $V(i)$.
- Partial knowledge about the data is used to build barrier potential, encoded in the matrix V, to stir the diffusion process in order to obtain suitable results

[8] W. Czaja and M. Ehler, Schroedinger eigenmaps for the analysis of biomedical data, IEEE Transactions on Pattern Analysis and Machine Intelligence 35 (2013), no. 5, 1274-1280.

Schroedinger eigenmaps: algorithm

Given a set of n points $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ in $\mathbb{R}^d$ and a function μ,

$$\mu : X \to \mathbb{R},$$

containing the extra information over the set of points X, the SE algorithm we will be using in our work involves the following steps:

- **Step 1:** Construct the adjacency graph.
- **Step 2:** Define a graph Laplacian, L, using the weight matrix, W.
- **Step 3:** Define the Schrodinger matrix, S, using the extra information, μ.

$$S = L + \alpha V,$$

where α is a real number, and V is the diagonal potential matrix given by

$$V = \begin{bmatrix} \mu_1 & & & \\ & \mu_2 & & \\ & & \ddots & \\ & & & \mu_n \end{bmatrix}, \tag{4}$$

where $\mu_i = \mu(\mathbf{x}_i)$ for all $i = 1, \ldots, n$.

Schroedinger eigenmaps: algorithm (continues)

- **Step 4:** Find the m-dimensional mapping by solving the generalized eigenvector problem,

$$S\mathbf{f} = \lambda D\mathbf{f}, \tag{5}$$

where $\mathbf{f}$ is a vector in $\mathbb{R}^n$ and λ is a real number. Let $\{\mathbf{f}_0, \mathbf{f}_1, \ldots, \mathbf{f}_{n-1}\}$ be the solution set to (5) written in ascending order according to their eigenvalues $\{\lambda_0, \lambda_1, \ldots, \lambda_{n-1}\}$. The m-dimensional Euclidean space mapping is given by

$$\mathbf{x}_i \rightarrow [\mathbf{f}_1(i), \mathbf{f}_2(i), \ldots, \mathbf{f}_m(i)].$$

Outline

1 Overview

2 Introduction

3 Transport by advection

4 Experiments and results

Continuous model: notations

We consider a graph as a set of points $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ in $\mathbb{R}^d$, or equivalently as a set of indices i in $I = \{1, 2, \ldots, n\}$.

- We denote A_i as the set of adjacent indices to i
- We denote $A = \{(i, j) : j \in A_i\}$ as the set of edges of the graph
- We denote $\mathcal{P}$ as the set of probability distributions μ from I to $\mathbb{R}^+$, μ is such that

$$\mu \in \mathcal{P} \Rightarrow \sum_i \mu_i = 1.$$

- We denote $\mathcal{E}$ as the set of functions from A to $\mathbb{R}$, and
- We denote $\mathcal{E}_a$ as the set of functions v in $\mathcal{E}$ that are antisymmetric, that is

$$v_{ij} = -v_{ji}.$$

Continuous model: definition

Let $\mu \in \mathcal{P}$, and $v \in \mathcal{E}_a$ a velocity field that is itself a function of μ. The transport model we consider is also known as the transport by advection; it refers to the active transportation of a distribution, μ, by a flow field, v.

We define the transport operator, T, acting on μ as follows:

$$T\mu = \triangle\mu - div(v\mu). \tag{6}$$

- $\triangle$ denotes the Laplacian defined as the divergence of the gradient acting on a distribution μ.
- *div* denotes the divergence, a vector operator that produces a scalar field quantifying a vector field's source at each point.
- We denote ∇ as the gradient acting on a scalar field.
- A comprehensive study of the operator in (6) is found in related materials by Benamou et al.[9] and Hundsdorfer et al.[10].
- Given an appropriately chosen flow field, v, we are able to direct the diffusion process in order to form desirable clusters.

[9] J-D. Benamou, B. D. Froese, and A. M. Oberman, Numerical solution of the optimal transportation problem using the Monge-Ampère equation, Journal of Computational Physics 260 (2014), 107-126.

[10] W. Hundsdorfer and J. G. Verwer, Numerical solution of time-dependent advection-diffusion-reaction equations, vol. 33, Springer Science & Business Media, 2013.

Discrete model: discretization

We propose the following discretization as well as matrix formulation:

- Given a function μ on I, we define the gradient of μ as $\nabla\mu$ by

$$(\nabla\mu)_{ij} = w_{ij}(\mu_j - \mu_i).$$

- We also define the Laplacian of μ as $\triangle\mu = div(\nabla\mu)$ by

$$(\triangle\mu)_i = \sum_{j\in A_i} w_{ij}(\mu_j - \mu_i).$$

- The centered discretization of $v\mu$ is given by:

$$(v\mu)^c_{ij} = v_{ij}\frac{\mu_i + \mu_j}{2}.$$

Our choice of discretization schemes is motivated by its well-defined analytic properties.

Discrete model and derivative

We consider a purely local type of flow by taking $v = \beta \nabla \mu$, where β is a real number. Using the central discretization, we obtain the following equation:

$$\begin{aligned}(T\mu)_i &= \sum_{j \in A_i} w_{ij}(\mu_j - \mu_i) - \beta \sum_{j \in A_i} w_{ij}(\mu_j^2 - \mu_i^2), \quad \text{for each} \quad i \in I \\ &= (F_l(\mu))_i - \beta (F_d(\mu))_i, \quad \text{for each} \quad i \in I.\end{aligned}$$

The derivatives of F_l and F_d with respect to μ are:

$$F_l'(\mu) = L \quad \text{and} \quad F_d'(\mu) = 2C_\mu \circ L,$$

where the operation $\circ$ is the element-wise multiplication, and the matrix C_μ is given by

$$C_\mu = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ \mu_1 & \mu_2 & \dots & \mu_n \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}.$$

Linearization

We finally write the linearization, $\tilde{T}$, of the transport operator around any given distribution μ as

$$\tilde{T}(u) = [L - 2\beta C_\mu \circ L](u) \quad \forall u \in \mathcal{P}. \tag{7}$$

Transport eigenmaps algorithm

Given a set of n points $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ in $\mathbb{R}^d$ and a function μ

$$\mu : X \to \mathbb{R}$$

over the set of points X. The transport eigenmaps algorithm involves the following steps:

- **Step 1:** Construct the adjacency graph.
- **Step 2:** Define a graph Laplacian, L, using the weight matrix, W.
- **Step 3:** Define the linearized transport matrix, $\tilde{T}$, using the extra information, μ.

  $$\tilde{T} = L - 2\beta C_\mu \circ L,$$

  where β is a real number, the operation $\circ$ is the element-wise multiplication and

$$C_\mu = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ \mu_1 & \mu_2 & \ldots & \mu_n \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}, \tag{8}$$

  where $\mu_i = \mu(\mathbf{x}_i)$ for all $i = 1, \ldots, n$.

Transport eigenmaps algorithm (continues)

- **Step 4:** Find the m-dimensional mapping by solving the generalized eigenvector problem,

$$\tilde{T}\mathbf{f} = \lambda D\mathbf{f}, \tag{9}$$

where $\mathbf{f}$ is a vector in $\mathbb{R}^n$ and λ is a real number. Let $\{\mathbf{f}_0, \mathbf{f}_1, \ldots, \mathbf{f}_{n-1}\}$ be the solution set to (9) written in ascending order according to their eigenvalues $\{\lambda_0, \lambda_1, \ldots, \lambda_{n-1}\}$. The m-dimensional Euclidean space mapping is given by

$$\mathbf{x}_i \rightarrow [\mathbf{f}_1(i), \mathbf{f}_2(i), \ldots, \mathbf{f}_m(i)].$$

Outline

1 Overview

2 Introduction

3 Transport by advection

4 Experiments and results

Example: Laplacian, transport, and Schroedinger mapping

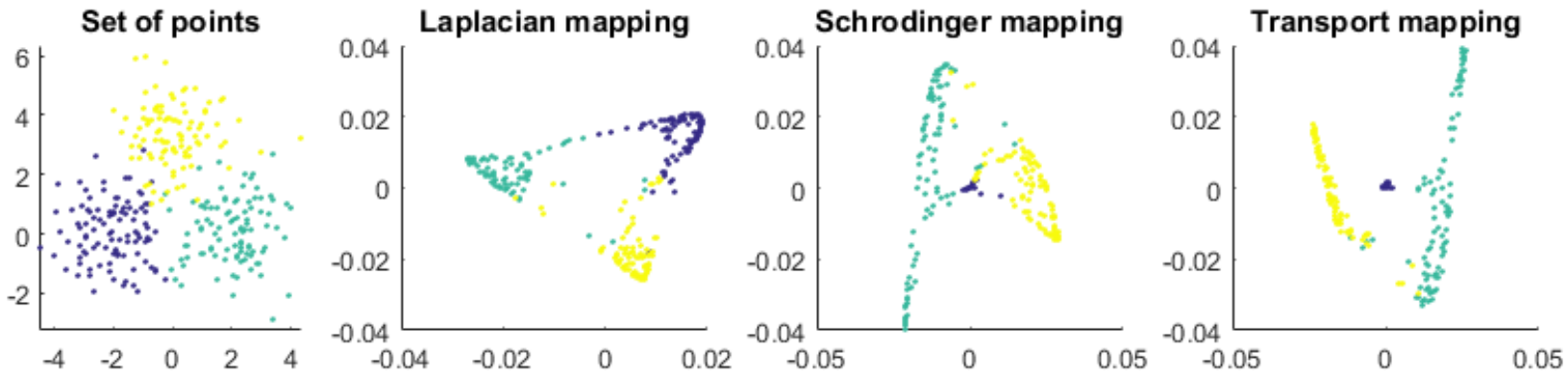


Figure 2: The first plot represents a set of points, 300 points grouped in 3 clusters of 100 points each, in the order blue, green, then yellow. The rest represents various mapping using the first and second eigenvectors with corresponding non-zero eigenvalue.

The adjusted Rand index (ARI)

Given a set X of n points and two partitions, e.g., clusterings, of these points, viz., $P = \{P_1, P_2, \ldots, P_r\}$ and $Q = \{Q_1, Q_2, \ldots, Q_s\}$, the adjusted Rand index[11][12] is defined as

$$ARI = \frac{\sum_{ij} \binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}}, \tag{10}$$

where $n_{ij} = |P_i \cap Q_j|$, $a_i = |P_i|$, and $b_j = |Q_j|$, for $i = 1, \ldots, r$ and $j = 1, \ldots, s$.

The adjusted Rand index:

- Quantifies the similarities between two clusterings from 0 to 1.
- A 0 indicates that the clusterings do not agree on any pair of points and a 1 indicates that the clusterings are exactly the same.
- ARI is robust against random chance assignments.

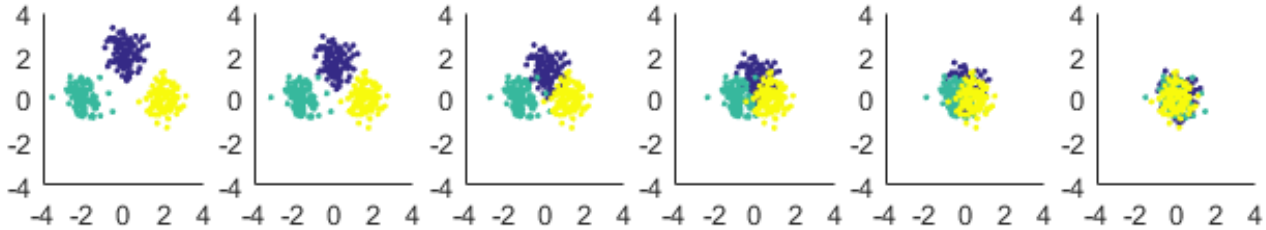
[11] W. M. Rand, Objective criteria for the evaluation of clustering methods, Journal of the American Statistical Association, 66.336, 1971, pp. 846-850.

[12] J. M. Santos and M. Embrechts, On the use of the adjusted Rand index as a metric for evaluating supervised classification, International Conference on Artificial Neural Networks, Springer, 2009, pp. 175-184.

Representation experiment: setup

We demonstrate the strength of our algorithm in its ability to faithfully represent the data using a large number of experiments.

- Clusters are arranged in 2 or 3 dimensions increasing in difficulty.
- We increase the difficulty by changing the parameters used to generate the data set: position, spread or standard deviation, added Gaussian noise, and number of clusters.
- Example: changing the position



- We use the adjusted Rand index (ARI) to quantify the representation of the data set.

Representation experiment: procedure

For each individual run, the following operations are performed:

- **Step 1:** Generate the data set, X, and corresponding labels.
- **Step 2:** We cluster the data set before and after dimension reduction using the k-means algorithm.
- **Step 3:** We compute the adjusted Rand index before and after dimension reduction and store the difference.

The following dimension reduction algorithms are used in the experiment:

- Principal components analysis (PCA),
- Laplacian eigenmaps (LE),
- Diffusion maps (DIF),
- Isomap (ISO),
- Schroedinger eigenmaps (SE),
- Transport eigenmaps (TE).

Representation experiment: result - overall

A positive change in ARI implies a better representation of the data after dimension reduction:

- The rightmost position of the box plot corresponding to TE in relation to the other DR algorithms implies that in general, TE produces the best representation of the data.

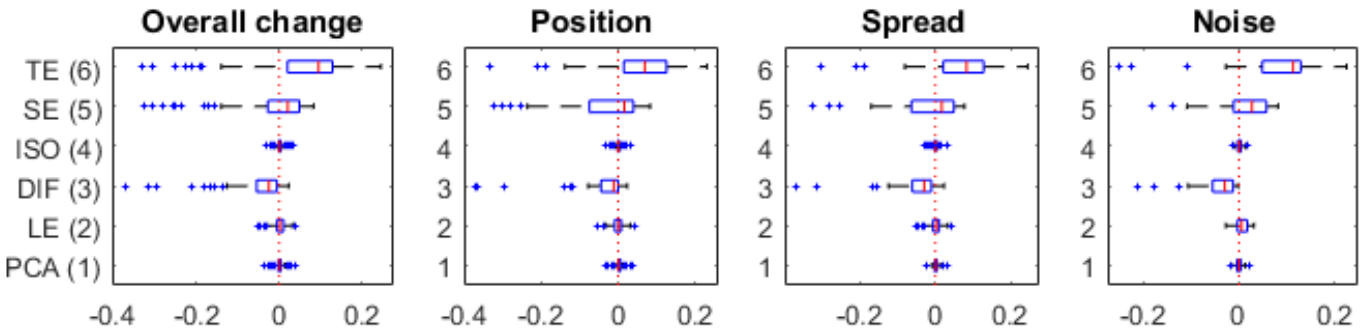


Figure 3: Box plot for the change of ARI, all $162 \times 20 = 3240$ cases.

Representation experiment: result - complex and simple

The dominant performance of TE is more apparent on difficult cases, see Figure 4, than it is on simple cases, see Figure 5.

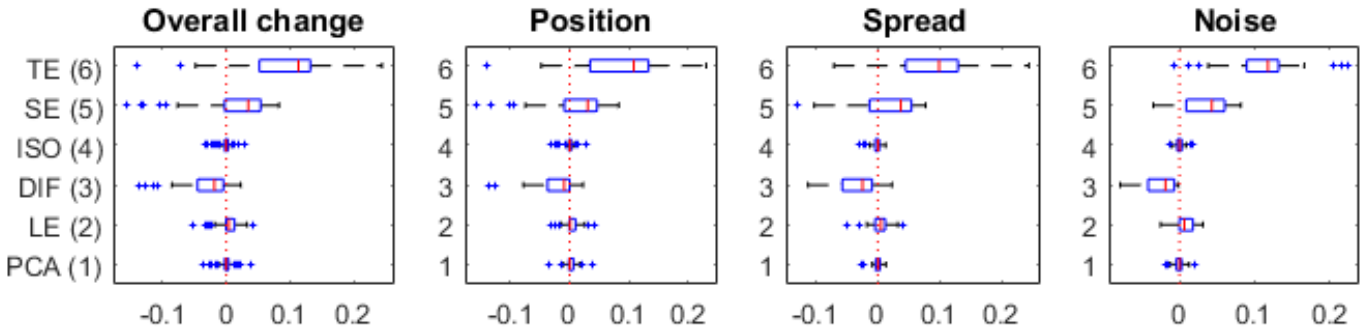


Figure 4: Box plot for the change of adjusted Rand index, $126 \times 20 = 2520$ difficult cases.

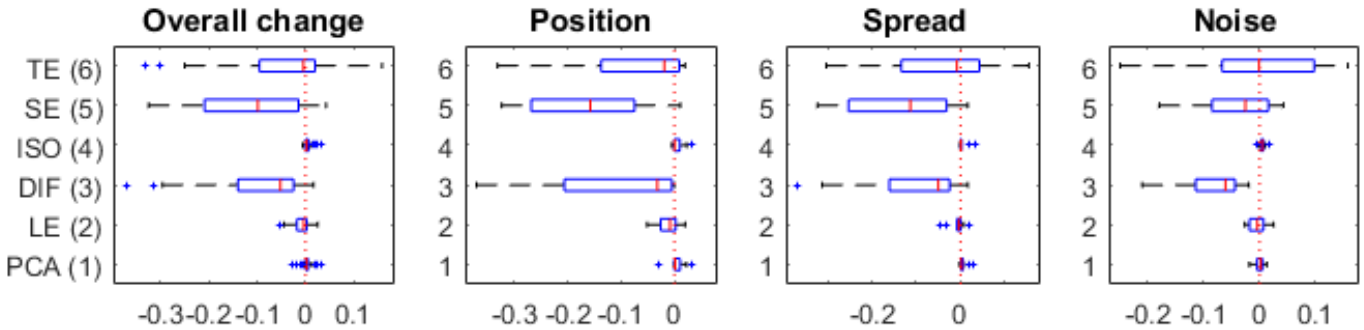


Figure 5: Box plot for the change of adjusted Rand index, $36 \times 20 = 720$ simple cases.

Hyperspectral dataset: Indian Pines

In this section, we work with the Indian Pines[13][14] data set:

- Gathered by AVIRIS (Airborne Visible/Infrared Imaging Spectrometer) sensor.
- Over the Indian Pines test site in North-western Indiana: $145 \times 145 \times 200$.
- Hyperspectral bands covering the region of water absorption have been removed.

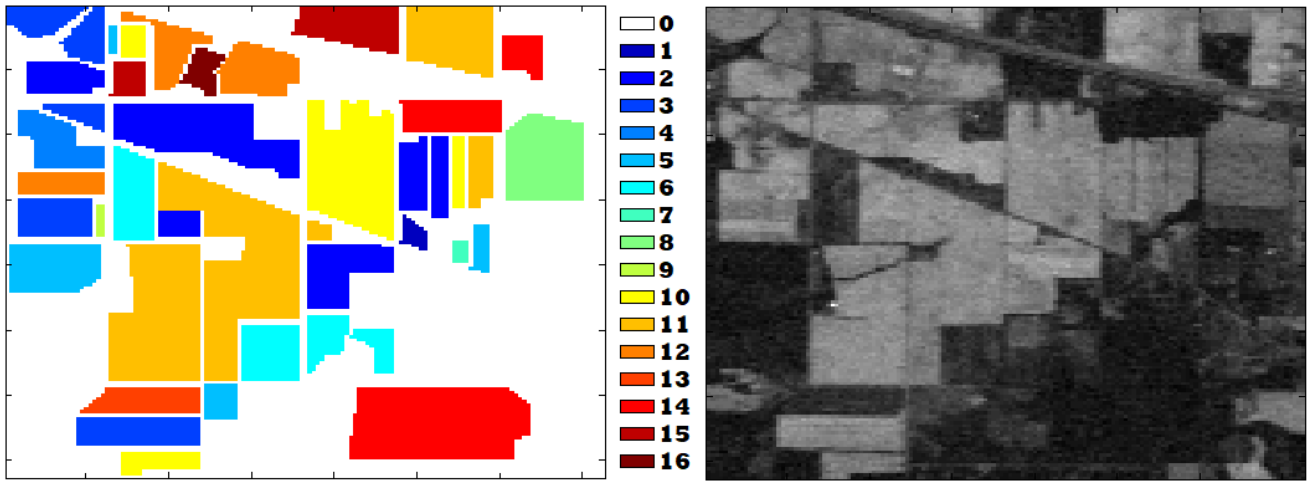


Figure 6: Ground truth (left) and sample band: 170 (right.)

[13] M. F. Baumgardner, L. L. Biehl, and D. A. Landgrebe, 220 band AVIRIS hyperspectral image data set: June 12, 1992 Indian Pines test site 3, September 2015.

[14] Hyperspectral remote sensing scenes, http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes, Accessed: 2018-04-04.

Indian Pines: ground truth

#	Class	Sample
0	Empty-space	10776
1	Alfalfa	46
2	Corn-notill	1428
3	Corn-mintill	830
4	Corn	237
5	Grass-pasture	483
6	Grass-trees	730
7	Grass-pasture-mowed	28
8	Hay-windrowed	478
9	Oats	20
10	Soybean-notill	972
11	Soybean-mintill	2455
12	Soybean-clean	593
13	Wheat	205
14	Woods	1265
15	Buildings-Grass-Trees-Drives	386
16	Stone-Steel-Towers	93

Table 1: Indian Pines classes.

Indian Pines grouped: ground truth

#	Class	Sample
0	Empty-space	10776
1	Alfalfa	46
2	Corn	2495
5	Grass	1241
8	Hay-windrowed	478
9	Oats	20
10	Soybean	4020
13	Wheat	205
14	Woods	1265
15	Buildings-Grass-Trees-Drives	386
16	Stone-Steel-Towers	93

Table 2: Indian Pines-G classes, ground truth with corresponding grouped labels.

Extra information and parameters

- Given prior knowledge about class 11–soybean-mintill in the Indian Pines data set, we would place a potential for SE or an advection for TE on class 11–soybean-mintill using the function μ defined as follows:

$$\mu(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{x} \in \text{Class 11–soybean-mintill,} \\ 0, & \text{elsewhere.} \end{cases}$$

- We ran a set of experiments to obtain the following parameters:
  - We set $m = 50$ (Indian Pines), $k = 12$, and $\sigma = 1$.
  - We set $\beta = 10$ and $\hat{\alpha} = 10^4$, where the parameter $\hat{\alpha}$ such that $\alpha = \hat{\alpha} \cdot \text{tr}\,(L)/\text{tr}\,(V)$.

Classification and validation metric

After the embedding:

- We use the 1-nearest neighbor algorithm to classify the data sets.
- We use 10% of the data from each class to train the classifier and the rest, N_v, as the validation set.
- We took an average of ten runs to produce the confusion matrices, C.

We following validation metrics are reported:

- The adjusted Rand index (ARI) between the predicted labels and the ground truth.
- The overall accuracy (OA).
- The Cohen's kappa coefficient[15][16] (κ) is defined by

$$\kappa = \frac{N_v \sum_i (C_{i,i})^2 - \omega}{N_v^2 - \omega},$$

where $\omega = \sum_i C_{i,\cdot} C_{\cdot,i}$. Similar to ARI, κ measures the agreement between clusterings, a 0 indicates no agreement while a 1 indicates complete agreement.

[15] Smeeton, N. C., Early history of the kappa statistic, 1985, pp. 795-795.

[16] Galton, F, Finger Prints Macmillan, 1892.

Results: Indian Pines

The best representation and accuracy come from TE, with SE as a close second, see Table 3.

IP	PCA	LE	DIF	ISO	SE-2	SE-11	TE-2	TE-11
ARI	0.4426	0.3694	0.4210	0.3929	0.5520	0.6955	**0.5735**	**0.7085**
OA	0.6761	0.6081	0.6556	0.6308	0.7138	0.7353	**0.7281**	**0.7418**
κ	0.6301	0.5532	0.6065	0.5785	0.6732	0.6981	**0.6900**	**0.7055**

Table 3: Classification results for Indian Pines (IP).

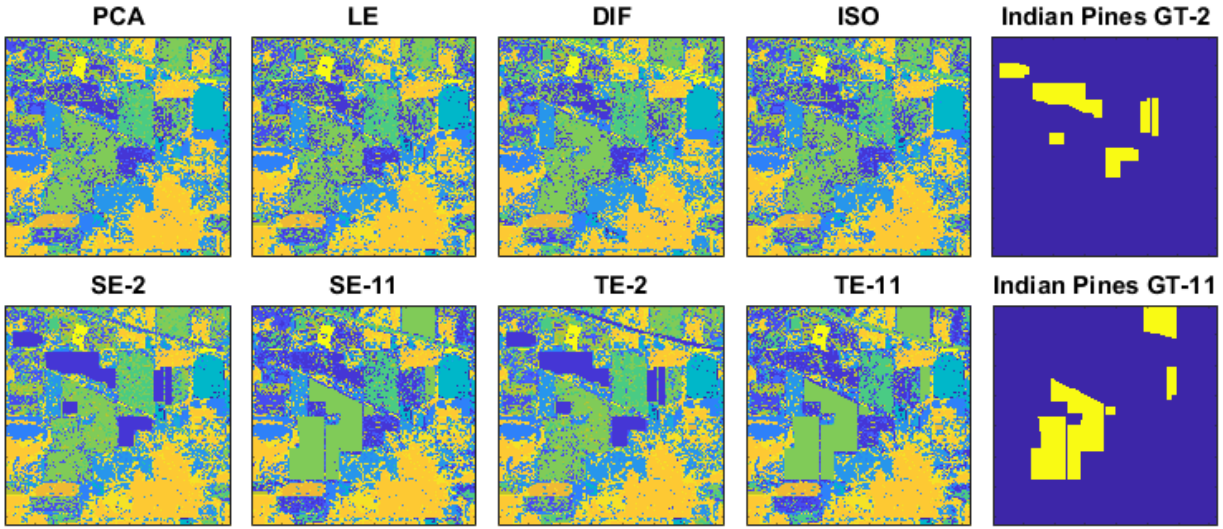


Figure 7: Classification map.

Results: TE vs SE vs % of information

- With lesser information provided from a particular class, SE is slightly ahead of TE.
- As a more complete information is provided, TE outperforms SE.

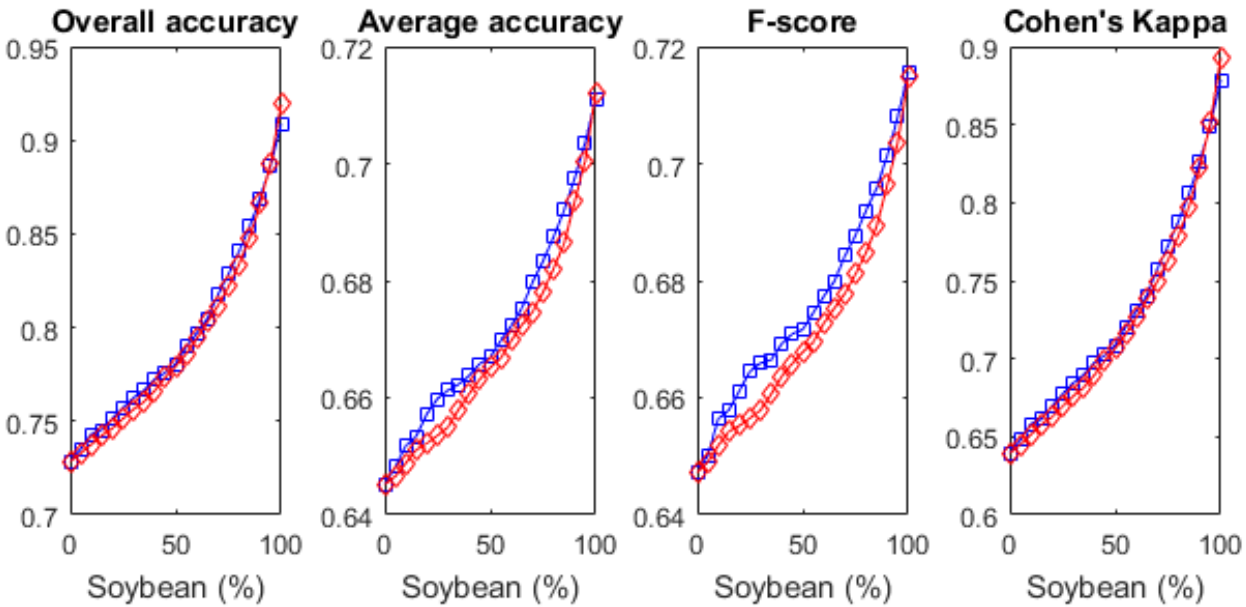


Figure 8: Classification performance measures for TE (red diamonds) and SE (blue squares) as a function of the amount of information provided. The Indian Pines-G data set is used with the advection and potential is placed on class 10–soybean.

Results: robustness against noise

- The added Gaussian noise has a mean of 0, we selected 20 logarithmically spaced values for the standard deviation from 10^0 to 10^5.
- In general, TE is the most robust algorithm against noise.

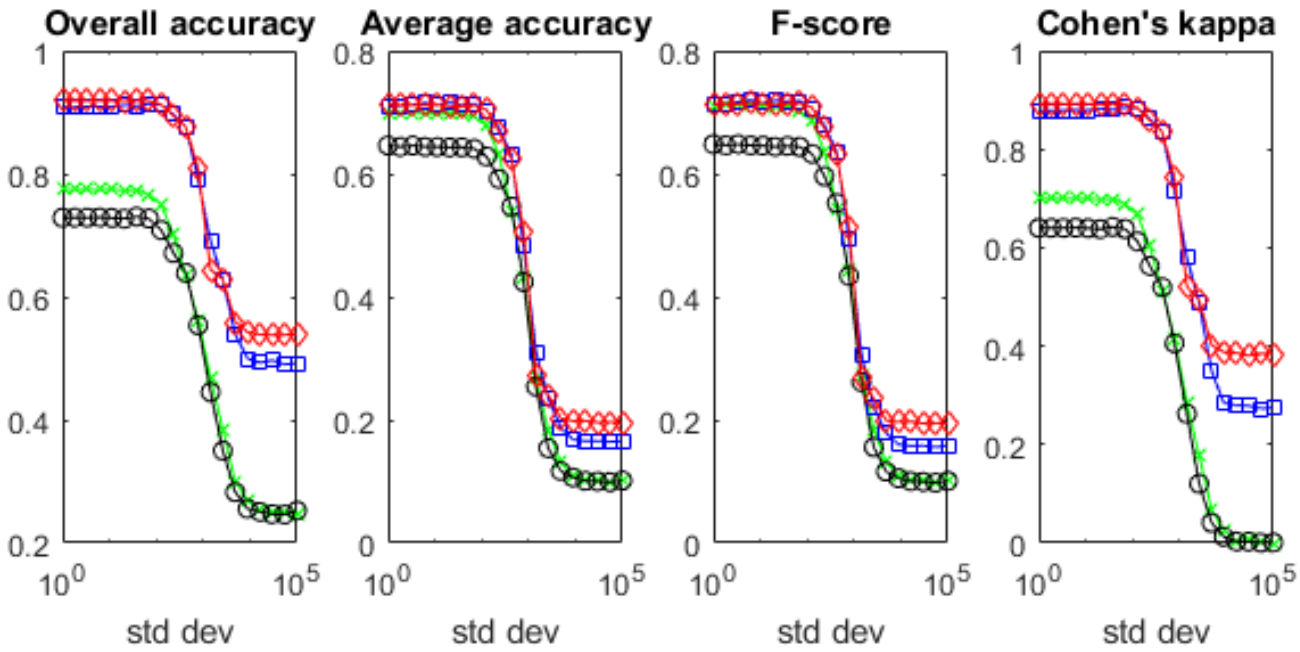


Figure 9: TE (red diamonds), SE (blue boxes), PCA (green x's), and LE (black circles). The Indian Pines-G data set is used with the advection and potential is placed on class 10–soybean.

Conclusion

- We constructed a novel semi-supervised non-linear dimension reduction algorithm based on a transport model by advection.
- We used advection, the active transportation of a distribution by a flow field, to stir the diffusion process in order to get better or more desirable results.
- We provided a set of experiments based on artificially generated data sets and on publicly available hyperspectral data set to show that our algorithm exhibits superior/competitive performance.
- We believe that the performance of our algorithm can be improved by choosing alternative flow fields and/or using different linearization techniques.

Thank You!