Low-Rank Representation over the Manifold of Curves

Stephen Tierney[1], Junbin Gao[2], Yi Guo[3] and Zhengwu Zhang[4]

[1]School of Computing and Mathematics, Charles Sturt University, Bathurst, NSW 2795, Australia

[2]Business Analytics Discipline, The University of Sydney Business School, Camperdown NSW 2006, Australia

[3]Digital Productivity & Services Flagship, CSIRO, North Ryde, NSW 2113, Australia

[4]Department of Statistics, Statistical and Applied Mathematical Sciences Institute, NC 27709-4006, USA

stierney@csu.edu.au; junbin.gao@sydney.edu.au; yi.guo@csiro.au; zhengwustat@gmail.com

Abstract

In machine learning it is common to interpret each data point as a vector in Euclidean space. However the data may actually be functional i.e. each data point is a function of some variable such as time and the function is discretely sampled. The naive treatment of functional data as traditional multivariate data can lead to poor performance since the algorithms are ignoring the correlation in the curvature of each function. In this paper we propose a method to analyse subspace structure of the functional data by using the state of the art Low-Rank Representation (LRR). Experimental evaluation on synthetic and real data reveals that this method massively outperforms conventional LRR in tasks concerning functional data.

1. Introduction

In machine learning it is common to interpret each data point as a vector in Euclidean space [3]. Such a discretisation is chosen because it allows for easy closed form solutions and fast computation, even with large datasets. However these methods ignore the fact that the data may not naturally fit into this assumption. In fact much of the data collected for practical machine learning are actually functions i.e. curves. For example financial data such as stock or commodity prices are functions of monetary value over time. Functional data have become increasingly important in many scientific and engineering research areas such as ECG (electrocardiogram) or EEG (Electroencephalography) in healthcare, biology data analysis, weather or climate data and motion trajectories from computer vision.

Analyzing functional data has been an emerging topic in statistical research [7, 13, 20, 21] and has attracted great attention from machine learning community in recent years [2, 15]. One of important challenges in analyzing functional data for machine learning is to efficiently cluster and

to learn better representations for functional data. Theoretically the underlying process for functional data is of infinite dimension, thus it is difficult to work with them with only finite samples available. A desired model for functional data is expected to properly and parsimoniously characterize the nature and variability hidden in the data. The classic functional principal component analysis (fPCA) [17] is one of such examples to discover dominant modes of variation in the data. However fPCA may fail to capture patterns if the functional data are not well aligned in its domain. For time series, a special type of functional data, dynamic time warping (DTW) has long been proposed to compare time series based on shape and distortions (e.g., shifting and stretching) along the temporal axis [16, 24].

Another important type of functional data is shape [23, 20]. Shape is an important characterizing feature for objects and in computer vision shape has been widely used for the purpose of object detection, tracking, classification, and recognition. In fact, a natural and popular representation for shape analysis is to parametrize boundaries of planar objects as 2D curves. In object recognition, images of the same object should be similar regardless of resolution, lighting, or orientation. Hence an efficient shape representation or shape analysis scheme must be invariant to scale, translation and rotation. A very useful shape representation is the square-root velocity function (SRVF) representation [9, 20]. In general, the resulting SRVF of a continuous shape is square integrable, a well-defined Hilbert space where appropriate measurement can be applied, refer to Section 2 for more details. By acknowledging the true nature of the data we can develop more powerful methods that exploit features that would otherwise be ignored or lead to erroneous results with simple linear models.

Our intention in this study is to consider functional data clustering by accounting for the possible invariance in scaling/stretching, translation and rotation of functional data to help maintain shape characteristics. The focus of this paper is upon functional data where data sets consist of continu-

ous real curves including shapes in Euclidean spaces. More specifically we propose a method of subspace analysis for functional data based on the idea developed in recent subspace clustering. The idea is to apply a feature mapping such as the aforementioned SRVF to the curves so that they are transformed onto the curve manifold, where the subspace analysis can be conducted based on the geometry on the manifold. In particular, we adapt the well known low-rank representation (LRR) framework [12] to deal with data that lie on the manifold of open curves by implementing the classical LRR in tangent spaces of the manifold [8, 25, 29].

LRR on Euclidean spaces [12] is closely related to several state-of-the-art subspace analysis approaches such as Sparse Subspace Clustering (SSC) [6], Robust PCA (RPCA) [5] and low-rank Matrix Completion (MC) [28] methods. This mixture of subspaces model has naturally led to the development of subspace segmentation methods. Such methods aim to segment the data into clusters with each cluster corresponding to a unique subspace. More formally, given a data matrix of observed column-wise data samples $\mathbf{A} = [\mathbf{a_1}, \mathbf{a_2}, \ldots, \mathbf{a_N}] \in \mathbb{R}^{D \times N}$, the objective of subspace clustering is to assign each data sample to its underlying subspace. The basic assumption is that the data within $\mathbf{A}$ is drawn from a union of c subspaces $\{S_i\}_{i=1}^c$ of dimensions $\{d_i\}_{i=1}^c$.

The core of both SSC and LRR is to learn an affinity matrix for the given dataset and the learned affinity matrix will be pipelined to a spectral clustering method like nCUT [19] to obtain the final subspace labels. To learn the affinity matrix, SSC relies on the self expressive property [6], which is that

each data point in a union of subspaces can be efficiently reconstructed by a linear combination of other points in the data.

In other words, each point can be written as a linear combination of the other points i.e. $\mathbf{A} = \mathbf{AZ}$, where $\mathbf{Z} \in \mathbb{R}^{N \times N}$ is a matrix of coefficients. Most methods however assume the data generation model $\mathbf{X} = \mathbf{A} + \mathbf{N}$, where $\mathbf{X}$ is the observed data and $\mathbf{N}$ is noise. Since it is difficult to separate the noise from the data the solution is to relax the self-expressive model to $\mathbf{X} = \mathbf{XZ} + \mathbf{E}$, where $\mathbf{E}$ is a fitting error and is different from $\mathbf{N}$.

Similarly LRR [12] exploits the self expressive property but attempts to learn the global subspace structure by computing the lowest-rank representation of the set of data points. In other words, data points belonging to the same subspace should have similar coefficient patterns. In the presence of noise LRR attempts to minimise the following objective

$$\min_{\mathbf{Z}, \mathbf{E}} \ \frac{1}{2} \|\mathbf{E}\|_\ell + \text{rank}(\mathbf{Z}), \quad \text{s.t.} \quad \mathbf{X} = \mathbf{XZ} + \mathbf{E}. \quad (1)$$

However rank minimisation is an intractable problem. Therefore LRR actually uses the nuclear norm $\|\cdot\|_*$ (sum of the matrix's singular values) as the closest convex relation

$$\min_{\mathbf{Z}, \mathbf{E}} \ \frac{1}{2} \|\mathbf{E}\|_\ell + \|\mathbf{Z}\|_*, \quad \text{s.t.} \quad \mathbf{X} = \mathbf{XZ} + \mathbf{E}, \quad (2)$$

where $\|\cdot\|_\ell$ is a placeholder for the norm most appropriate to the expected noise type. For example in the case of Gaussian noise the best choice is the ℓ_2 norm i.e. $\|\cdot\|_F^2$ and for sparse noise the ℓ_1 norm should be used.

Both SSC and LRR rely on the linear self expressive property. This property is no longer available in the nonlinear manifold, e.g. the manifold of open curves as mentioned previously. To generalize LRR or SSC for data in the manifold space, we explicitly explore the underlying nonlinear data structure and utilize the techniques of exponential and logarithm mappings to bring data to a local linear space.

The rest of the paper is organized as follows. In Section 2, we review the preliminaries about the manifold of open curves and introduce the curve Low-Rank Representation (cLRR) model. Section 3 is dedicated to explaining an efficient algorithm for solving the optimization proposed in cLRR based on the linearized alternative direction method with adaptive penalty (LADMAP) and the algorithm convergence and complexity are also analyzed. In Section 4, the proposed model is assessed on both synthetic and real world databases against several state-of-the-art methods. Finally, conclusions are discussed in Section 5.

2. LRR over the Curve Manifold

As previously discussed LRR is limited to a linear model and its current version can only be applied to vector data from a Euclidean space. Matrix $\mathbf{Z}$ in (1) or (2) encodes the affinity/similarity between data points. However this assumption is often unnatural and quite limiting. Much of the data encountered in real world is functional. In other words it exhibits a curve like structure over a domain. Euclidean linear models are unable to capture the nonlinear invariance embedded in each data point. For example in thermal infra-red data of geological substances a curve may contain a key identifying feature such as a dip near a particular frequency. This dip may shift or vary position over time even for the same substance due to impurities. Under a linear vector model this variation may cause the vector to drastically move in the ambient Euclidean space and cause poor results. Or in other cases the feature may be elongated, shrunk or be subject to some non-uniformly warping or scaling. In all these cases the linear model will fail to accurately represent the non-linear affinity in the data.

Exploring these unique non-linear invariance in functional data is the focus of this paper. We now discuss how to adapt LRR (similar approach appliable to SSC) such that it

easily accepts curve data and nonlinear relationships within clusters can be easily discovered.

2.1. The Curve Manifold

Given a smooth parameterized n-dimension curve $\beta : D = [0,1] \to \mathbb{R}^n$, we represent it using he square-root velocity function (SRVF) representation [9, 20], which is given by

$$q(t) = \frac{\dot{\beta}(t)}{\sqrt{\|\dot{\beta}(t)\|}}.$$

The SRVF mapping transforms the original curve $\beta(t)$ into a gradient based representation, which facilitates the comparing of the shape information.

In this paper, we focus on the set of open curves, e.g. the curves do not form a loop ($\beta(0) \neq \beta(1)$). For handling general curves, we refer readers to [20]. The SRVF facilitates a measure and geometry bearing invariance to scaling, shifting and reparameterization in the curves domain. For example, all the translated curves from a curve $\beta(t)$ will have the same SRVF. Robinson [18] proved that if the curve $\beta(t)$ is absolutely continuous, then its SRVF $q(t)$ is square-integrable, i.e., $q(t)$ is in a functional Hilbert space $L^2(D, \mathbb{R}^n)$. Conversely for each $q(t) \in L^2(D, \mathbb{R}^n)$, there exists a curve $\beta(t)$ whose SRVF corresponds to $q(t)$. Thus the set $L^2(D, \mathbb{R}^n)$ is a well-defined representation space of all the curves. The most important advantage offered by the SRVF framework is that the natural and widely used L^2-measure on $L^2(D, \mathbb{R}^n)$ is invariant to the reparameterization. That is, for any two SRVFs q_1 and q_2 and a randomly chosen reparametrization function (non-decreasing) $t = \gamma(\tau)$, we have

$$\|q_1(t) - q_2(t)\|_{L^2} = \|q_1(\gamma(\tau)) - q_2(\gamma(\tau))\|_{L^2}.$$

This property has been exploited in [2] for functional data clustering under the subspace clustering framework. Different from the work proposed in [2], we will adopt the newly developed LRR on manifolds framework to the model of curves LRR, see [8, 25, 29]. To see this, we introduce some more notation. Let Γ be the set of all diffeomorphisms from $D = [0,1]$ to $D = [0,1]$. This set collects all the reparametrization mappings. Γ is a Lie group with the composition as the group operation and the identity mapping as the identity element. Then all the orbits $[q] = \{q \circ \gamma = q(\gamma(t)) \mid \forall \gamma \in \Gamma\}$ together define the quotient manifold $L^2(D, \mathbb{R}^n)/\Gamma$.

Without loss of generality, all curves are normalized to have unit length, i.e., $\int_D \|\dot{\beta}(t)\|dt = 1$. The SRVFs associated with these curves are elements of a unit hypersphere in the Hilbert space $L^2(D, \mathbb{R}^n)$, i.e., $\int_D \|q(t)\|^2 dt = 1$. Therefore, under the curve normalization assumption, in-

stead of $L^2(D, \mathbb{R}^n)$, we consider the following unit hypersphere manifold

$$\mathcal{C}^o = \left\{ q \in L^2(D, \mathbb{R}^n) : \int_D \|q(t)\|^2 dt = 1 \right\}.$$

The manifold $\mathcal{C}^o$ has some nice properties, see [1]. For any two points q_0 and q_1 in $\mathcal{C}^o$, a geodesic connecting them is given by $\alpha : [0,1] \to \mathcal{C}^o$,

$$\alpha(\tau) = \frac{1}{\sin(\theta)}(\sin(\theta(1-\tau))q_0 + \sin(\theta\tau)q_1), \quad (3)$$

where $\theta = \cos^{-1}(\langle q_0, q_1 \rangle)$ is the length of the geodesic. If we take derivative of α w.r.t to q_1, the tangent vector at q_0 is

$$v = \frac{\theta}{\sin(\theta)}[q_1 - \langle q_0, q_1 \rangle q_0]. \quad (4)$$

The above formula is regarded as the $\widehat{\text{Logarithm}}$ mapping $\log_{q_0}(q_1)$ on the manifold $\mathcal{C}^o$.

As we are concerned with the shape invariance, i.e., we need to additionally remove the shape-preserving transformations: rotation and curve reparametrization. The manifold concerning us is the quotient space of the manifold $\mathcal{S}^o = \mathcal{C}^o/(SO(n) \times \Gamma)$, where $SO(n)$ is the rotation group. Each element $[q] \in \mathcal{S}^o$ is an equivalent class defined by

$$[q] = \left\{ Oq(\gamma(t))\sqrt{\dot{\gamma}(t)} \mid O \in SO(n) \text{ and } \gamma \in \Gamma \right\}.$$

Given any two points $[q_0]$ and $[q_1]$ in $\mathcal{S}^o$, a tangent representative [1] in the tangent space $T_{[q_0]}(\mathcal{S}^o)$ can be calculated in the following way, as suggested in [31, 22] based on (4),

$$\widetilde{v} = \log_{q_0}(\widetilde{q}_1) = \frac{\widetilde{\theta}}{\sin(\widetilde{\theta})}[\widetilde{q}_1 - \langle q_0, \widetilde{q}_1 \rangle q_0]. \quad (5)$$

where $\widetilde{q}_1$ is the representative of $[q_1]$ given by the well-defined algorithm in [31, 22] and $\widetilde{\theta} = \cos^{-1}(\langle q_0, \widetilde{q}_1 \rangle)$. In fact, $\widetilde{v}$ is the lifting representation of abstract tangent vector $\log_{[q_0]}([q_1])$ on $T_{[q_0]}(\mathcal{S}^o)$ at q_1.

2.2. The Proposed Curve LRR

Given a set of N unit-length curves $\{\beta_1(t), ..., \beta_N(t)\}$, denote their SRVFs by $\{q_1(t), ..., q_N(t)\}$ such that $[q_i] \in \mathcal{S}^o$ and $q_i(t)$ is a representative of the equivalent class $[q_i]$. We cannot apply the standard LRR model (2) directly on the quotient manifold $\mathcal{S}^o$. This is because (2) indeed relies on the following individual linear combination

$$\mathbf{x}_i = \sum_{j=1}^{N} z_{ij}\mathbf{x}_j + \mathbf{e}_i, \quad (6)$$

which is invalid for $[q_i]$'s on $\mathcal{S}^o$. Note that z_{ij} can be explained as the affinity or similarity between data points $\mathbf{x}_i$ and $\mathbf{x}_j$.

On any manifold, the tangent space at a given point is linearly local approximation to the manifold around the point and the linear combination is valid in the tangent space. This prompts us to replace the affinity relation in (6) by the following

$$\log_{[q_i]}([q_i]) = \sum_{j=1}^{N} w_{ij} \log_{[q_i]}([q_j]) + \mathbf{e}_i \qquad (7)$$

with the constraint $\sum_{j=1}^{N} w_{ij} = 1, i = 1, 2, \ldots, N$ to maintain consistency at different locations. The meaning of w_{ij} in (7) is the similarity between curves $\beta_i(t)$ and $\beta_j(t)$ via the "affinity" between tangent vectors $\log_{[q_i]}([q_i])$ and $\log_{[q_i]}([q_j])$ at the first order approximation accuracy. Each $\log_{[q_i]}([q_j])$ can be calculated by (5) and it is obvious that $\log_{[q_i]}([q_i]) = 0$ for any i.

With all the ingredients at hand, we are fully equipped to propose the curve LRR (cLRR) model as follows

$$\min_{\mathbf{W}} \lambda\|\mathbf{W}\|_* + \sum_{i=1}^{N} \frac{1}{2}\|\sum_{j=1}^{N} w_{ij} \log_{[q_i]}([q_j])\|_{[q_i]}^2,$$

$$\text{s.t. } \sum_{j=1}^{N} w_{ij} = 1, i = 1, 2, \ldots, N. \qquad (8)$$

where $\|\cdot\|_{[q]}$ is the metric defined on the manifold, which is defined by the classic L^2 Hilbert metric on the tangent space.

Denote $\mathbf{w}_i$ the i-th row of matrix $\mathbf{W}$ and define

$$B_{jk}^i = \langle \log_{[q_i]}([q_j]), \log_{[q_i]}([q_k])\rangle. \qquad (9)$$

Then with some algebraic manipulation we can re-write the model (8) into the following simplified form,

$$\min_{\mathbf{W}} \lambda\|\mathbf{W}\|_* + \sum_{i=1}^{N} \mathbf{w}_i \mathbf{B}^i \mathbf{w}_i^T,$$

$$\text{s.t. } \sum_{j=1}^{N} w_{ij} = 1, i = 1, 2, \ldots, N. \qquad (10)$$

where $\mathbf{B}^i = (B_{jk}^i)$.

Effectively this objective allows for similarity between curves to be measured in their tangent spaces. Our highly accurate segmentation results in Section 4 have demonstrated that this is an effective way to learn non-linear similarity.

3. Optimisation

3.1. Algorithm

To solve the cLRR objective we use the Linearized Alternative Direction Method with Adaptive Penalty

(LADMAP) [10, 11] . First take the Augmented Lagrangian of the objective (10)

$$L = \lambda\|\mathbf{W}\|_* + \frac{1}{2}\sum_{i=1}^{N} \mathbf{w}_i \mathbf{B}^i \mathbf{w}_i^T + \langle \mathbf{y}, \mathbf{W1} - \mathbf{1}\rangle$$
$$+ \frac{\beta}{2}\|\mathbf{W1} - \mathbf{1}\|_F^2 \qquad (11)$$

where $\mathbf{y}$ is the Lagrangian multiplier (vector) corresponding to the equality constraint $\mathbf{W1} = \mathbf{1}$, $\|\cdot\|_F$ is the matrix Frobebius-norm, and we will update β as well in the iterative algorithm to be introduced.

Denote by $F(\mathbf{W})$ the function defined by (11) except for the first term $\lambda\|\mathbf{W}\|_*$. To solve (11), we adopt a linearization of $F(\mathbf{W})$ at the current location $\mathbf{W}^{(k)}$ in the iteration process, that is, we approximate $F(\mathbf{W})$ by the following linearization with a proximal term

$$F(\mathbf{W}) \approx F(\mathbf{W}^{(k)}) + \langle \partial F(\mathbf{W}^{(k)}), \mathbf{W} - \mathbf{W}^{(k)}\rangle$$
$$+ \frac{\eta_W \beta_k}{2}\|\mathbf{W} - \mathbf{W}^{(k)}\|_F^2,$$

where η_W is an approximate constant with a suggested value given by $\eta_W = \max\{\|B_i\|^2\} + N + 1$, and $\partial F(\mathbf{W}^{(k)})$ is a gradient matrix of $F(\mathbf{W})$ at $\mathbf{W}^{(k)}$. Denote by $\mathbf{B}$ the 3-order tensor whose i-th front slice is given by $\mathbf{B}^i$. Let us define $\mathbf{W} \odot \mathbf{B}$ the matrix whose i-row is given by $\mathbf{w}_i \mathbf{B}^i$, then it is easy to show

$$\partial F(\mathbf{W}^{(k)}) = \mathbf{W} \odot \mathbf{B} + \mathbf{y1}^T + \beta_k(\mathbf{W1} - \mathbf{1})\mathbf{1}^T. \quad (12)$$

Then (11) can be approximated by linearization and $\mathbf{w}$ will be updated by the following

$$\mathbf{W}^{(k+1)} = \arg\min_{\mathbf{W}} \lambda\|\mathbf{W}\|_* \qquad (13)$$

$$+ \frac{\eta_W \beta_k}{2}\left\|\mathbf{W} - \left(\mathbf{W}^{(k)} - \frac{1}{\eta_W \beta_k}\partial F(\mathbf{W}^{(k)})\right)\right\|_F^2.$$

Problem (13) admits a closed form solution by using SVD thresholding operator [4], given by

$$\mathbf{W}^{(k+1)} = U_W S_{\frac{\lambda}{\eta_W \beta_k}}(\Sigma_W)V_W^T, \qquad (14)$$

where $U_W \Sigma_W V_W^T$ is the SVD of $\mathbf{W}^{(k)} - \frac{1}{\eta_W \beta_k}\partial F(\mathbf{W}^{(k)})$ and $S_\tau(\cdot)$ is the Singular Value Thresholding (SVT) [4, 14] operator defined by

$$S_\tau(\Sigma) = \text{diag}(\max\{|\Sigma_{ii}| - \tau, 0\}). \qquad (15)$$

The updating rule for $\mathbf{y}$

$$\mathbf{y}^{(k+1)} = \mathbf{y}^{(k)} + \beta_k(\mathbf{W}^{(k)}\mathbf{1} - \mathbf{1}) \qquad (16)$$

and the updating rule for β_k

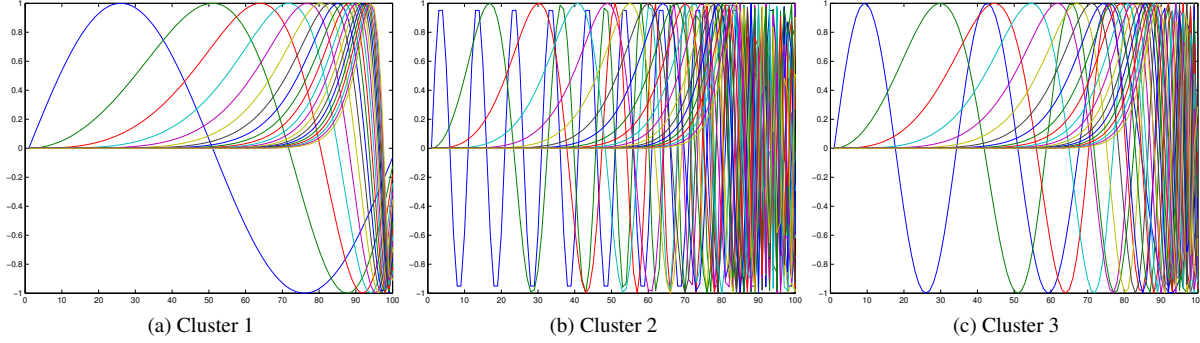$$\beta_{k+1} = \min\{\beta_{\max}, \rho\beta_k\}, \qquad (17)$$

Figure 1: Example plots of curves generated in the Synthetic Data Experiment. Each cluster has a base sine curve (the left most blue curve) which is progressively warped with each successive instantiation.

Algorithm 1 Solving (10) by LADMAP

Require: $\{\mathbf{X}_i\}_{i=1}^N$, λ
1: Initialise: $\mathbf{W} = \mathbf{0}$, $\mathbf{y} = \mathbf{0}$, $\beta = 0.1$, $\beta_{\max} = 10$, $\rho^0 = 1.1$, $\eta = \max\{\|\mathbf{B}^i\|_F\} + N + 1$, $\epsilon_1 = 1e^{-4}$, $\epsilon_2 = 1e^{-4}$
2: Construct each $\mathbf{B}^i$ as per (9)
3: **while** not converged **do**
4:    Update $\mathbf{W}$ using (14)
5:    Check convergence criteria

$$\beta^{(k)}\|\mathbf{W}^{(k+1)} - \mathbf{W}^{(k)}\|_F \le \epsilon_1$$
$$\|\mathbf{W1} - \mathbf{1}\|_F \le \epsilon_2$$

6:    Update Lagrangian Multiplier

$$\mathbf{y}^{(k+1)} = \mathbf{y}^k + \beta^{(k)}(\mathbf{W1} - \mathbf{1})^T$$

7:    Update ρ

$$\rho = \begin{cases} \rho_0 & \text{if } \beta^{(k)}\|\mathbf{W}^{(k+1)} - \mathbf{W}^{(k)}\|_F \le \epsilon_1 \\ 1 & \text{otherwise,} \end{cases}$$

8:    Update β

$$\beta^{(k+1)} = \min(\beta_{\max}, \rho\beta^{(k)})$$

9: **end while**
10: **return** $\mathbf{W}$

---

where

$$\rho = \begin{cases} \rho_0 & \beta_k\|\mathbf{W}^{k+1} - \mathbf{W}^k\| \le \varepsilon_1, \\ 1 & \text{otherwise.} \end{cases}$$

We summarize the above as Algorithm 1. Once the coefficient matrix $\mathbf{W}$ is found, a spectral clustering like nCUT [19] is applied on the affinity matrix $\frac{|\mathbf{W}|+|\mathbf{W}|^T}{2}$ to obtain the segmentation of the data.

3.2. Complexity Analysis

For ease of analysis, we firstly define some symbols used in the following. Let K and r denote the total number of iterations and the lowest rank of the matrix $\mathbf{W}$, respectively. The size of $\mathbf{W}$ is $N \times N$. The major computation cost of our proposed method contains two parts, calculating all $\mathbf{B}^i$'s and updating $\mathbf{W}$. In terms of the formula (9) through (4) and (5), the computational complexity of Log algorithm is $O(T^2)$ where T is the number of terms in a discretized curves; therefore, the complexity of B_{jk}^i is at most $O(T^2)$ and $\mathbf{B}^i$'s computational complexity is $O(N^2T^2)$. Thus the total for all the $\mathbf{B}^i$ is $O(N^3)$. In each iteration of the Algorithm, the singular value thresholding is adopted to update the low rank matrix $\mathbf{W}$ whose complexity is $O(rN^2)$ [12]. Suppose the algorithm is terminated after K iterations, the overall computational complexity is given by

$$O(N^3) + O(KrN^2)$$

3.3. Convergence Analysis

Algorithm 1 is adopted from the algorithm proposed in [11]. However due to the terms of $\mathbf{B}^i$'s in the objective function (11), the convergence theorem proved in [11] cannot be directly applied to this case as the linearization is implemented on both the augmented Lagrangian terms and the term involving $\mathbf{B}^i$'s. Fortunately we can employ the revised approach, presented in [30], to prove the convergence for the algorithm. Without repeating all the details, we present the convergence theorem for Algorithm 1 as follows.

Theorem 1 (Convergence of Algorithm 1) *If*

$$\eta_W \ge \max\{\|B_i\|^2\} + N + 1, \quad \sum_{k=1}^{+\infty} \beta_k^{-1} = +\infty,$$

$$\beta_{k+1} - \beta_k > C_0 \frac{\sum_i \|B_i\|^2}{\eta_W - \max\{\|B_i\|^2\} - N}, \text{ where } C_0 \text{ is a}$$
given constant and $\| \cdot \|$ is the matrix spectral norm, then

the sequence $\{W^k\}$ generated by Algorithm 1 converges to an optimal solution to problem (10).

In all the experiments we have conducted, the algorithm converges very fast with $K < 100$.

4. Experiments

In this section we show three sets of experiments to evaluate the newly proposed cLRR. The performance of the proposed method is compared with the same type of subspace clustering algorithm LRR [12]. To compare segmentation accuracy we use the subspace clustering accuracy (SCA) metric [6], which is defined as

$$\text{SCA} = 1 - \frac{\text{num. of misclassified points}}{\text{total num. of points}}. \quad (18)$$

Therefore a higher SCA % means greater clustering accuracy.

The parameters used were fixed across all experiments with λ for LRR set at 1 and 0.1 for cLRR. A wide range of parameters were tested for each algorithm. Overall we found that the segmentation accuracy of LRR did not vary that much with changes in λ.
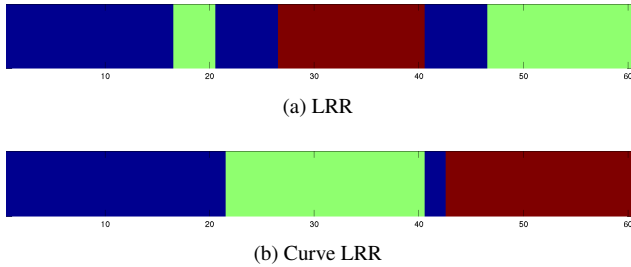
4.1. Synthetic Data

(a) LRR

(b) Curve LRR

Figure 2: The segmentation results from the data in Figure 1.

	Mean	Median	Min	Max
LRR	80.4%	83.33%	60%	91.67%
CurveLRR	**96.77%**	**98.33%**	**73.33%**	**100%**

Table 1: Synthetic Results

To evaluate and confirm the effectiveness of the curve LRR method we first perform experimental evaluation using synthetic data. In this test three clusters were created consisting of twenty 1-D curves of length 100. The curves in each cluster were sine waves, with each cluster corresponding to a unique frequency. Within each cluster progressive amounts of warping were applied. See Figure 1 for

an example of data from three syntheticly generated clusters. Clustering was then performed on the data by applying curve LRR and segmenting the affinity matrix with nCUT. This experiment was repeated 50 times with new data generated each time to obtain basic statistics. We compare against the baseline: LRR. Results are reported using subspace clustering accuracy and can be found in Table 1. Overall in this experiment Curve LRR outperforms conventional LRR by a significant margin.

4.2. Semi-synthetic TIR Data

We assemble synthetic data from a library of pure infrared hyper spectral mineral data. For each cluster we pick one spectra sample from the library as a basis. Each curve basis is then randomly shifted and stretched in a random portion. This random warping is performed 20 times to produce the curves for each cluster. See Figure 3 for an example of data used in this experiment. In this experiment we used three clusters. Again as in the previous experiment we repeated the test 50 times. Results are reported in Table 2 and Figure 4.

The results show that LRR cannot accurately cluster data with this sort of nonlinear invariance, which is commonly found in this type of data due to impurities in the mineral samples. On the other hand cLRR perfectly clustered the data.
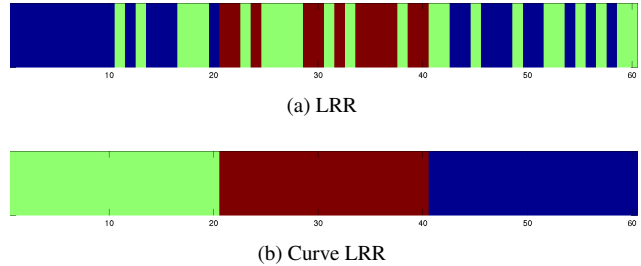

(a) LRR

(b) Curve LRR

Figure 4: The segmentation results from the data in Figure 3.

	Mean	Median	Min	Max
LRR	60.13%	60%	50%	71.67%
CurveLRR	**100%**	**100%**	**100%**	**100%**

Table 2: Semi-Synthetic TIR Results

4.3. Character Classification

In this experiment a collection of handwritten English characters were used to evaluate performance on a real world data set. The dataset consists of pen position data collected by a digitisation tablet at 200Hz, which is then converted to horizontal and vertical velocities [27, 26]. These
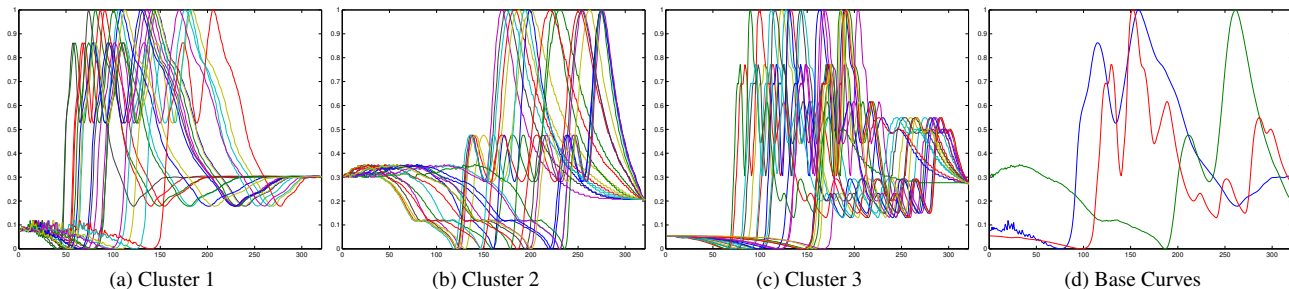
Figure 3: Example plots of curves used in the Semi-synthetic TIR Data Experiment. Each cluster has a base curve from the TIR library. The curves for each cluster have been shifted and stretched randomly from the base.
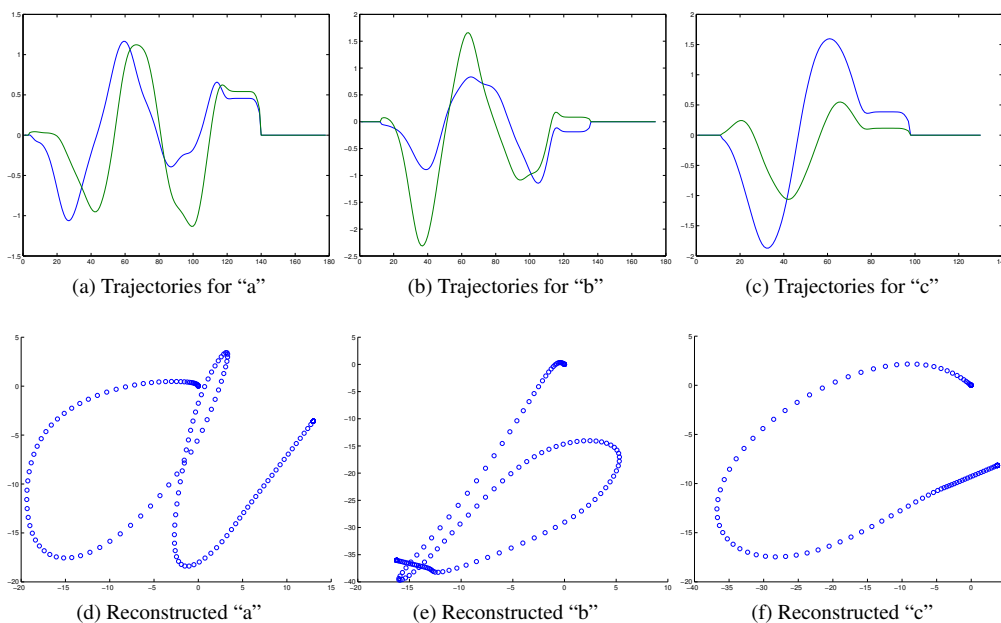

Figure 5: Example data from the character classification dataset. The top row plots the x and y pen tip velocities over time for three sample characters. The bottom row shows the corresponding character reconstruction by integrating the pen tip velocity data (for visualisation only).

2-D trajectory curves are normalised such that the mean of each curve is close to zero. See Figure 5 for some examples of this data. Figure 6 shows the example plots of curves used in the character classification experiment.

To evaluate performance twenty characters were randomly selected from three character classes. The data as originally released has been carefully produced and processed so that trajectories for each characters are extremely similar. Far more so than is realistic. For example the start time for each character has been aligned furthermore the writing speed, character size and variance in velocity over time is extremely consistent. Therefore to make the data more realistic we randomly globally shift each character so that their start times vary. Furthermore we randomly glob-

ally stretch and shrink each trajectory to account for different writing speeds, we also scale the trajectories by applying constant factors to account for character size and we lastly perform local warping (as done in the semi-synthetic experiment) to account for variance in speed over time.

Since the data consists of multidimensional curves the X and Y trajectory curves were concatenated to form data usable for conventional LRR since it can only handle vectors. Results can be found in Table 3 and Figure 7. Once again, the cLRR clearly outperforms LRR in all metrics. Furthermore cLRR shows excellent performance with a median accuracy of over 90% on an extremely challenging dataset.
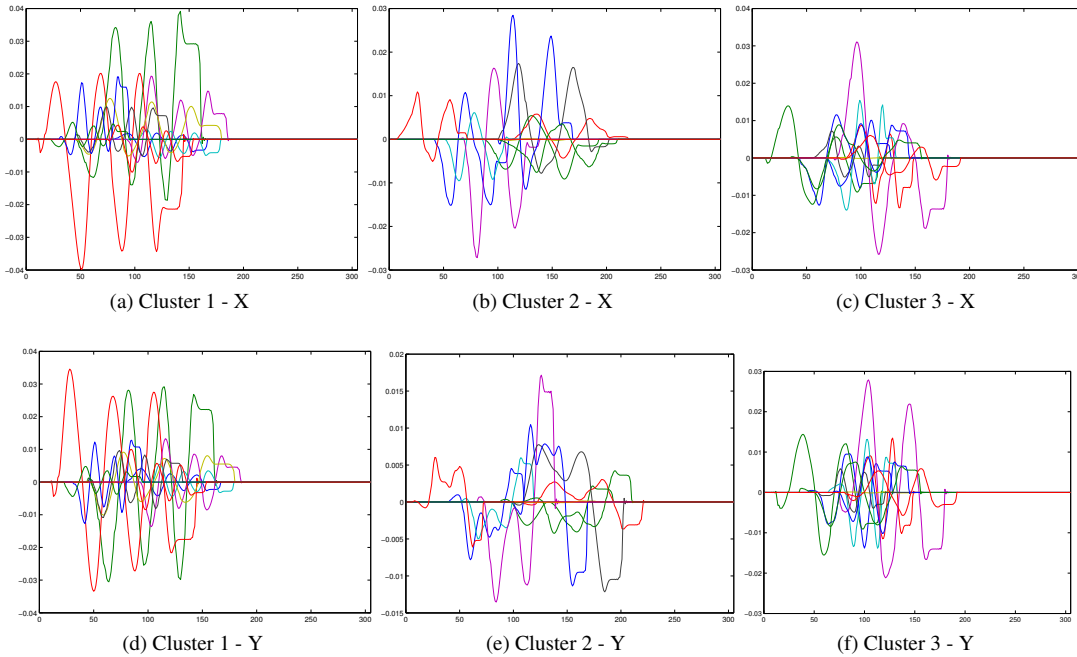
Figure 6: Example plots of curves used in the Character Classification Experiment. Each cluster consists of randomly selected characters from each class that are then subject to a combination of shifting, warping, stretching or shrinking and scaling. The top row shows the curves from the pen tip velocity in the X direction over time and the bottom row shows the same but for the Y direction.
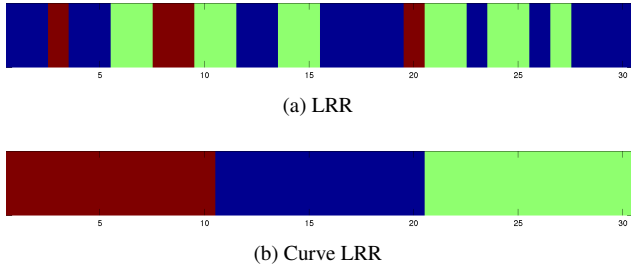

Figure 7: The segmentation results from the data in Figure 6.

	Mean	Median	Min	Max
LRR	52.33%	51.67%	43.33%	63.33%
CurveLRR	**86.33%**	**91.67%**	**70%**	**100%**

Table 3: Character Classification Results

5. Conclusion

In this paper, we extended the conventional LRR model on Euclidean space to a new LRR model for the manifold of open curves. The new LRR formulation is based on the tangent space approximation to the manifold so that the classic data self expressive can be well preserved for the manifold of curves at relevant high accuracy. The resulting optimization problem can be solved using the LADMAP technique and algorithm convergence and complexity were presented. Finally we tested the new model by conducting experiments on synthetic, semi-synthetic and real world data, and the experimental results show the outstanding performance against the conventional LRR. Our next work is further extended the LRR model to the manifold of general closed curves.

Acknowledgments

References

[1] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2008. 3

[2] M. T. Bahadori, D. Kale, Y. Fan, and Y. Liu. Functional subspace clustering with application to time series. In *Proceedings of The 32nd International Conference on Machine Learning*, pages 228–237, 2015. 1, 3

[3] C. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer, 2006. 1

[4] J. F. Cai, E. J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM J. on Optimization*, 20(4):1956–1982, 2008. 4

[5] E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? Submitted for publication, Stanford University, 2010. http://www-stat.stanford.edu/~candes/papers/RobustPCA.pdf. 2

[6] E. Elhamifar and R. Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2765–2781, 2013. 2, 6

[7] F. Ferraty and Y. Romain, editors. *The Oxford Handbook of Functional Data Analysis*. Oxford University Press, 2011. 1

[8] Y. Fu, J. Gao, X. Hong, and D. Tien. Low rank representation on Riemannian manifold of symmetrical positive definite matrices. In *SIAM Conferences on Data Mining (SDM)*, pages 316–324, 2015. 2, 3

[9] S. H. Joshi, E. Klassen, A. Srivastava, and I. Jermyn. A novel representation for Riemannian analysis of elastic curves in r^n. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7, 2007. 1, 3

[10] Z. Lin, R. Liu, and H. Li. Linearized alternating direction method with parallel splitting and adaptive penalty for separable convex programs in machine learning. *Machine Learning*, 99:287–325, 2015. 4

[11] Z. Lin, R. Liu, and Z. Su. Linearized alternating direction method with adaptive penalty for low rank representation. In *Proceedings of NIPS*, 2011. 4, 5

[12] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma. Robust recovery of subspace structures by low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):171–184, 2013. 2, 5, 6

[13] H.-G. Müller. *International Encyclopedia of Statistical Science*, chapter Functional data analysis, pages 554–555. Springer, 2011. 1

[14] N. Parikh and S. Boyd. Proximal algorithms. *Foundations and Trends in Optimization*, 1(3):123–231, 2013. 4

[15] F. Petitjean, G. Forestier, G. I. Webb, A. E. Nicholson, Y. Chen, and E. Keogh. Dynamic time warping averaging of time series allows faster and more accurate classification. in icdm, 2014. In *International Conference on Data Mining*, 2014. 1

[16] T. Rakthanmanon. Addressing big data time series: Mining trillions of time series subsequences under dynamic time warping. *ACM Transactions on Knowledge Discovery from Data*, 7(3):1–31, 2013. 1

[17] J. Ramsay and B. W. Silverman. *Functional Data Analysis*. Springer Series in Statistics. Springer, 2005. 1

[18] D. Robinson. *Functional analysis and partial matching in the square root velocity framework*. PhD thesis, Florida State University, 2012. 3

[19] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:888–905, 2000. 2, 5

[20] A. Srivastava, E. Klassen, S. H. Joshi, and I. H. Jermyn. Shape analysis of elastic curves in Euclidean spaces. *IEEE Transactionson Pattern Analysis and Machine Intelligence*, 33(7):1415–1428, 2011. 1, 3

[21] A. Srivastava, W. Wu, S. Kurtek, E. Klassen, and J. S. Marron. Registration of functional data using Fisher-Rao metric. *varXiv:1103.3817*, 2011. 1

[22] J. Su and A. Srivastava. Rate-invariant analysus of trajectories on Riemannian manifolds with application in visual speech recognition. In *Proceedings of International Conference on Computer Vision and Pattern Recognition*, 2014. 3

[23] J. Su, A. Srivastava, and F. W. Huffer. Detection, classification and estimation of individual shapes in 2D and 3D point clouds. *Computational Statistics & Data Analysis*, 58:227–241, 2013. 1

[24] J. D. Tucker, W. Wu, and A. Srivastava. Generative models for functional data using phase and amplitude separation. *Computational Statistics and Data Analysis*, 61:50–66, 2013. 1

[25] B. Wang, Y. Hu, J. Gao, Y. Sun, and B. Yin. Low rank representation on grassmann manifolds: An extrinsic perspective. *arXiv:1301.3529*, 1:1–9, 2015. 2, 3

[26] B. Williams, M. Toussaint, and A. J. Storkey. Modelling motion primitives and their timing in biologically executed movements. In *Advances in neural information processing systems*, pages 1609–1616, 2008. 6

[27] B. H. Williams, M. Toussaint, and A. J. Storkey. *Extracting motion primitives from natural handwriting data*. Springer, 2006. 6

[28] L. Wu, A. Ganesh, B. Shi, Y. Matsushita, Y. Wang, and Y. Ma. Convex optimization based low-rank matrix completion and recovery for photometric stereo and factor classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, XX:XXX–XXX, August 2012. 2

[29] M. Yin, J. Gao, and Y. Guo. A nonlinear low-rank representation on Stiefel manifold. *Electronics Letters*, 51(10):749–751, 2015. 2, 3

[30] M. Yin, J. Gao, Z. Lin, Q. Shi, and Y. Guo. Graph dual regularized low-rank matrix approximation for data representation. *IEEE Transactions on Image Processing*, 24(12):4918–4933, 2015. 5

[31] Z. Zhang, J. Su, E. Klassen, H. Le, and A. Srivastava. Video-based action recognition using rate-invariant analysis of covariance trajectories. *arXiv:1503.06699v1*, 1, 2015. 3