July 2018

Transfer Learning with Mixtures of Manifolds

Thomas Boucher

TRANSFER LEARNING WITH MIXTURES OF MANIFOLDS

A Dissertation Presented

by

THOMAS BOUCHER

Submitted to the Graduate School of the
University of Massachusetts Amherst in partial fulfillment
of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2018

College of Information and Computer Sciences

TRANSFER LEARNING WITH MIXTURES OF MANIFOLDS

A Dissertation Presented

by

THOMAS BOUCHER

Approved as to style and content by:

Sridhar Mahadevan, Chair

M. Darby Dyar, Member

Erik Learned-Miller, Member

Daniel Sheldon, Member

James Allan, Chair
College of Information and Computer Sciences

ACKNOWLEDGMENTS

I would like to thank my advisor Sridhar Mahadevan for providing a fountain of ideas and wisdom throughout the years. He welcomed me into UMass and his outstanding lab to explore my interests, and he taught me to enjoy the rigor of machine learning. At the start of my second year I met my other advisor Darby Dyar, and by the end of the year she had taken me from Amherst to Mars. I would like to thank Darby for always helping in the details, for exploring a new domain of machine learning with me, and especially for her mentorship and friendship.

I would like to thank my committee members, Erik Learned-Miller and Dan Sheldon, and the rest of the faculty for providing me a firm base to build my computer science research career.

I would like to thank the staff of the College of Information and Computer Sciences, especially Leeanne Leclerc and Susan Overstreet. Thank you to Susan, who kept my career on track and kept me paid (for quite a long time). Thank you to Leeanne, who helped me innumerable times, before I was accepted to UMass and during every semester since.

I would like to thank the members of the Autonomous Learning Laboratory for giving me a stimulating and comforting home during graduate school. I worked especially close (and lived) with some colleagues throughout the years, thank you to Luke Vilnis, CJ Carey, Peter Krafft, Steve Giguere, Phil Thomas, Ian Gemp, Stefan Dernbach, Francisco Garcia, and Clemens Rosenbaum. I would like to thank all of my friends outside of UMass too, whose encouragement, humor, and camaraderie have been absolutely crucial to me on my graduate school journey.

I would like to thank my whole family, especially my parents Anne and Tom. They raised me in a loving home and have encouraged me continuously throughout a winding course of life paths. I would like to thank my sisters, Juliana Caruso and Regina Fitek, for paving the way, helping me through, and putting me in my place, and I would like to thank their families, Jason, Alexandra, Matthew, and Dan, Jamie, Evan, who I am so grateful to have in my family.

During graduate school I married my wife Amanda. Most of all, I would like to thank Mandy a thousand times over. She has supported me in every way; she is my deepest love and my closest friend. Most important to this dissertation, she made me finish.

Lastly (since they cannot read), I would like to thank my cats Ehrie and Obie, a couple of great little distractors and comforters.

ABSTRACT

TRANSFER LEARNING WITH MIXTURES OF MANIFOLDS

MAY 2018

THOMAS BOUCHER

B.Sc., UNIVERSITY OF MASSACHUSETTS AMHERST

M.Sc., UNIVERSITY OF TENNESSEE KNOXVILLE

M.Sc., UNIVERSITY OF MASSACHUSETTS AMHERST

Ph.D., UNIVERSITY OF MASSACHUSETTS AMHERST

Directed by: Professor Sridhar Mahadevan

Advances in scientific instrumentation technology have increased the speed of data acquisition and the precision of sampling, creating an abundance of high-dimensional data sets. The ability to combine these disparate data sets and to transfer information between them is critical to accurate scientific analysis. Many modern day instruments can record data at many thousands of channels, far greater than the actual degrees of freedom in the sample data. This makes manifold learning, a class of methods that exploit the observation that high-dimensional data tend to lie on lower-dimensional manifolds, especially well-suited to this transfer learning task.

Existing manifold-based transfer learning methods can align related data sets in differing feature representations, but their inherent single manifold assumption causes them to fail in the presence of complex mixtures of manifolds. In this dissertation, a

new class of transfer learning algorithms is developed for high-dimensional data sets that intrinsically lie on multiple low-dimensional manifolds. With a more realistic mixture of manifolds assumption, this class of algorithms allows for accurate and efficient transfer of information between data sets by aligning their complex underlying geometries.

In this dissertation, algorithms are presented that leverage corresponding samples between data sets and available label information, continuous or categorical. The two primary tasks are aligning mixtures of manifolds and heterogeneous domain adaptation of multi-manifold data sets. Linear, non-linear, and robust versions of the algorithm are described, as well as a method for actively selecting cross-data set correspondences. To show the practical effectiveness of these algorithms, they are compared across a number of synthetic and real-world domains, but most notably to align data recorded by spectroscopic instruments during space exploration, a new domain for transfer learning.

TABLE OF CONTENTS

LIST OF TABLES

Table **Page**

LIST OF FIGURES

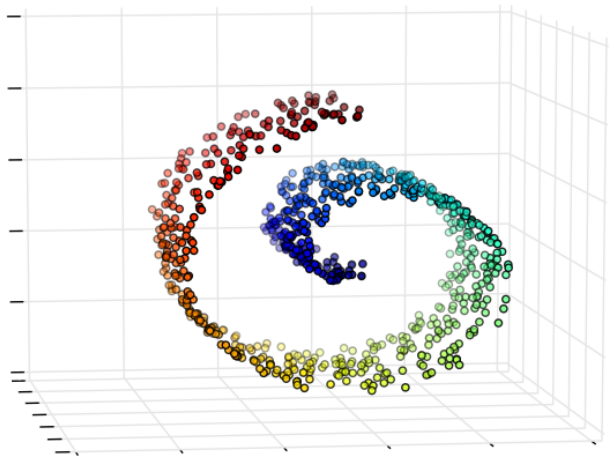
Figure **Page**

CHAPTER 1

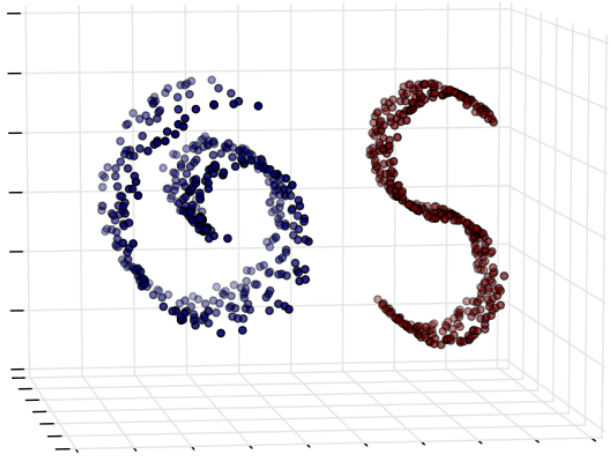
INTRODUCTION

1.1 Background and Motivation

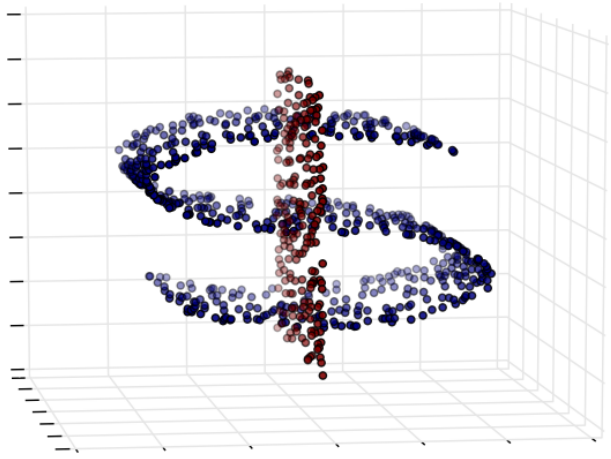
As machine learning practitioners tackle increasingly large and complex data sets, better data representations are necessary to improve task performance while reducing computational burden. This need is especially pervasive in data collected using modern scientific instrumentation, where advances in technology have increased the speed of data acquisition and the precision of sampling, creating an abundance of high-dimensional data sets. For example, instead of spectrometers recording at micrometer (10^{-6}) wavelength intervals, it is common to now record at picometer (10^{-12}) wavelength intervals. This type of trend occurs throughout many domains in machine learning, such as natural language processing, information retrieval, and bioinformatics. Manifold learning methods that exploit the observation that high-dimensional data tend to lie on lower-dimensional manifolds have proven to be especially useful in combating the *curse of dimensionality*, the demand for more training samples to fit a model in higher-dimensional space [5, 55, 65, 76].



(a) Single "Swiss roll" manifold.



(b) Non-overlapping distinct manifolds.



(c) Mixture of manifolds.

Figure 1.1: Types of manifold data.

One drawback of existing manifold learning approaches is their assumption that all data are drawn from one or more non-overlapping manifolds, as seen in Figure 1.1 a & b. This assumption stems from the use of distance-weighted local neighborhoods for graph construction, a technique that fails when data are drawn from an intersecting mixture of manifolds. As data sets begin to use representations that employ thousands or millions of features [54], the assumption of non-mixing manifolds becomes increasingly tenuous. This mixture of manifolds problem is demonstrated in Figure 1.1 c: when input manifolds are poorly separated, local neighborhood information is insufficient for recovering the true structure at manifold junctions. Thus, the use of traditional nearest neighbor graph construction algorithms induces incorrect connections at these intersection points, known as *short-circuits*, distorting the manifold representation. Figure 1.2 shows how both k-nearest neighbors and ϵ-ball approaches cause short-circuits between two non-intersecting manifolds that are simply nearby in space.

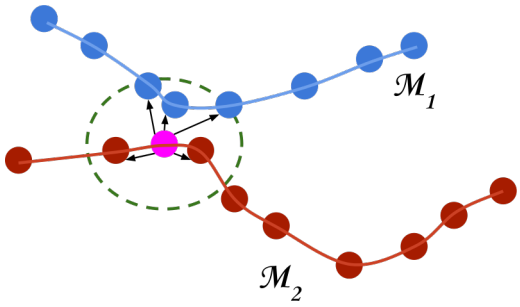


Figure 1.2: Using nearest neighbor with $k = 5$, the magenta point induces a *short-circuit* between nearby manifolds $\mathcal{M}_1$ and $\mathcal{M}_2$.

Another class of algorithms from dimensionality reduction that is related to manifold methods is subspace estimation [26, 29, 79]. In this paradigm, the data are assumed to be drawn from a set of underlying, typically low-dimensional, linear subspaces. These methods are attractive because instead of assuming a single underlying space, like traditional manifold methods, they are designed to tease apart multiple

underlying subspaces into their constituent parts. Many of these methods have focused on the task of clustering, whereby data are segmented based upon membership in particular underlying subspaces [26, 27, 71]. One limitation of these techniques is that they assume linear subspaces. The work in this dissertation is motivated by the idea that subspace estimation techniques can be used to empower traditional manifold learning algorithms to accomplish the task of multi-manifold estimation.

Transfer learning has especially benefited from manifold-based approaches, through continued advancements in manifold alignment [3, 41, 62, 69, 82, 84]. Data set alignment is a semi-supervised task where correspondences are learned between multiple data sets based on intra-set geometry and a provided partial set of pairwise correspondences between the data sets. Manifold alignment is a class of techniques that solves the data set alignment problem when the sets are assumed to share a common underlying structure, embedding each input set into a shared latent manifold space. Data set alignment would greatly benefit from mixture of manifold modeling. With a divide and conquer approach, the alignment problem is made simpler by first separating each data set to its constituent subspaces. The correspondences act as anchor points to orient the different manifolds between the data sets.

Data set alignment is unsupervised except for the known correspondences, but a related supervised transfer learning task is heterogeneous domain adaptation (HDA) [23, 83, 97]. HDA seeks to train a model using multiple related *source* data sets, in possibly very different ambient feature representations, for classification or regression of a related *target* data set, where training data are minimal. This is different from traditional domain adaptation, where the features are shared across source and target sets and are assumed to be nearby in distribution space. HDA solves a much broader set of problems, and like manifold alignment would equally benefit from mixture of manifold modeling.

1.2 Contributions and Outline of the Dissertation

In this dissertation, a new class of transfer learning algorithms is described for high-dimensional data sets that intrinsically lie on a shared set of low-dimensional manifolds. Existing methods that assume a single manifold fail in the presence of mixtures of manifolds. With a more realistic multi-manifold assumption, this class of algorithms allows for accurate and efficient transfer of information between data sets by aligning their complex underlying geometries. The dissertation is composed of manifold alignment and domain adaptation algorithms. Instead of aligning or mapping all data into a single latent manifold space, potentially mixing unrelated data, the methods developed in this dissertation only align related sections of data across sets by separating them apart according to their underlying mixture of manifolds.

The first contribution of the dissertation is ***low rank alignment***, LRA (Chapter 3), an unsupervised multi-manifold alignment algorithm based upon a low rank reconstruction framework [9]. The input to the algorithm is a set of heterogeneous data sets, where each data set may be in dramatically different feature representations. Each data set is assumed to intrinsically lie on a set of low-dimensional manifold spaces. Additionally, samples present across multiple data sets, called cross-data set *correspondences*, are used to tie the disparate manifold spaces together. The method is unsupervised because it does not use label information about the data, though it relies on the overlapping correspondences to reason across data sets. For each sample in each data set, LRA calculates a low-dimensional representation in a joint space called an *embedding*, whereby the geometry of the joint space is semantically meaningful. For example, if the task is to align pictures of cats and dogs from different cameras and drawings, then in the joint space photos and drawings of cats ought to be close together and separate from those of dogs. More generally, an embedding is a function from the ambient space to the joint space whose domain is restricted to the training set. The low rank penalty captures the underlying subspaces

of the individual data sets, while the correspondences anchor the subspaces between data sets. It is primarily a two step algorithm, with an optional preprocessing step, where each step has a closed-form solution. It does not suffer from the sensitive nearest neighbor hyperparameter present in traditional manifold alignment, nor does it require prior knowledge of the number of manifolds.

The second contribution of the dissertation is a set of ***three extensions to the low rank alignment algorithm*** (Sections 3.3-3.5). In the first step of the standard algorithm, a reconstruction matrix is calculated for each data set, and all three extensions make use of these reconstruction matrices. First, a kernelized variant of the algorithm specifically designed to handle highly non-linear manifold subspaces is presented. This variant requires little overhead cost compared to the original algorithm, while providing a much higher degree of adaptiveness in its reconstruction of individual data sets. Second, a variant of the standard algorithm for the task of clustering is described. A simple post-processing step is performed to the reconstruction matrix to encourage samples from the same underlying space to cluster closely together in the joint space. The integrity of the global geometry of the data sets is sacrificed to encourage subspace partitioning in the joint space. This intermediate step adds little computational burden to the algorithm and is shown to successfully dissect the samples according to subspaces. Third, a variant of the low rank alignment algorithm is presented that can actively learn high-value correspondences. Depending on the geometry of the underlying spaces of the data, not all samples are equally useful as correspondences. The reconstruction matrices can be used to actively learn which samples in each data set would be most useful for the alignment. Correspondences are typically expensive to acquire, so the ability to actively select a small number of them instead of relying on an abundance of randomly selected correspondences is very useful.

The third contribution of the dissertation is ***a set of robust low rank alignment algorithms*** (Chapter 4). Real-world data are often complex and noisy. While low rank alignment is good at using only samples of the same underlying space for reconstruction, when there are many non-linear intersection points, small noisy short-circuits can develop in the low rank reconstruction matrix. To suppress these short-circuits, a sparsity term is added to the calculation of the reconstruction matrices. This is shown to outperform standard low rank alignment in many cases; however, the additional sparsity term requires the optimization problem be solves iteratively. An iterative algorithm using the alternating direction method of multipliers (ADMM) [11] is described.

The low rank alignment algorithm is still susceptible to noise in the ambient data representation and to outliers. To address these issues, a noise term can be directly modeled in the first step of the alignment algorithm, thereby completely removing the noise and outliers from the calculation of the embedding in the second step of the algorithm. Two types of error modeling are described, one intended more for general noise and one intended more for outlier detection. The error term forces this step to be solved iteratively, and an ADMM-based algorithm is detailed.

The fourth contribution of the dissertation is ***heterogeneous domain adaptation (HDA) of multi-manifold sets*** (Chapter 5). In addition to these data alignment algorithms, methods for domain adaptation are presented. There are three key differences between the alignment methods and these HDA methods: (1) these are supervised methods, i.e., they make use of label information; (2) they are designed with a subsequent task in mind, e.g., classification, regression; and (3) they provide a map (function) to the joint space for each input data set for samples outside the original training set. First in this section, the existing work on single manifold HDA [83] that uses categorical (class) labels is extended to the multi-manifold case. Next, a novel framework for using continuous (real-valued) labels for the task

of regression is described. Like the class label type, it is assumed that the data sets each share a common response surface (label space), and then the algorithm uses that information to reason between the data sets. The methods maximize the squared cross-correlation between the shared label space and each of the data sets while preserving correspondences between data sets. Like alignment, this is a two-step algorithm, where a closed-form solution is provided for each step. Finally, a non-linear version of the algorithm is constructed by kernelizing the algorithm. In addition to providing a non-linear mapping, the kernel formulation poses the problem in its dual formulation. Typically it is the case that the dimensionality of the ambient space is much larger than the number of samples, so the dual formulation is often more computationally efficient. The linear kernel can be used to take advantage of the dual formulation while keeping the mapping linear.

The fifth contribution of the dissertation is ***a new real-world domain and problem for transfer learning*** (Sections 2.5, 3.6.1, 4.3.3, 5.2.3). To show the practical merit of this class of multi-manifold alignment methods, their effectiveness is primarily evaluated using spectroscopic and chemical data acquired by the *Curiosity* rover on Mars or in support of the mission in labs on Earth. These data, known as spectra, have a high ambient dimensionality but likely have a very low intrinsic dimensionality, making them good candidates for manifold methods. By aligning spectra from disparate instruments and laboratories, correcting for differences found in data recorded on Earth and those recorded on another planet, more accurate and complete chemical and mineralogical models can be trained. Labeled samples used to calibrate the classification and regression models of instruments are known in the literature as *standards*. Standards are frequently shared between labs, and these can be used as correspondences in the alignment model, but they come at a high price because their characterization is expensive. The feature representations of spectra are energy values, where the features are ordered according to wavelength,

and thus the data are structured. These spectra are typically partially labeled, by either their scientific classification (hierarchical discrete labels) or their chemical composition (multi-task continuous labels). In addition to traditional alignment tasks, the task of *calibration transfer* (CT) is presented. CT solves the problem of transferring a calibration curve from one instrument or set of conditions to another using a calculated transfer function, without the need to resample the calibration standards on both instruments/conditions. In all spectroscopic applications, there is a need to ensure that possible differences in instruments, environment, or experimental conditions are mitigated or negated.

CHAPTER 2

BACKGROUND AND RELATED WORK

In this chapter, relevant related work is discussed, starting with a brief description of the grandparent of transfer learning methods, canonical correlation analysis (CCA). Next follows an overview of manifold alignment, a field intended to correct the shortcomings of CCA. The general manifold alignment algorithm is then explained in detail, because it is a building block for the novel algorithms in the dissertation. The existing work on mixtures of manifolds is described next, followed by a brief overview of domain adaptation to address the usage of multiple related data sets. Finally, a description of the dissertation's motivating case study is presented: spectroscopy in space exploration.

2.1 CCA

Canonical correlation analysis (CCA) [44] is perhaps the most popular transfer learning technique. Given two data sets $X \in \mathbb{R}^{N \times p}$ and $Y \in \mathbb{R}^{N \times q}$, CCA calculates the linear subspace that maximizes correlation between the two sets. Assuming the data sets are first mean-centered, CCA optimizes the function

$$\underset{f,g:\|f\|=1,\|g\|=1}{\text{maximize}} \ \text{corr}\left(Xf, Yg\right) = \underset{f,g:\|f\|=1,\|g\|=1}{\text{maximize}} \ \frac{f^\top X^\top Y g}{\sqrt{\left(f^\top X^\top X f\right)\left(g^\top Y^\top Y g\right)}}.$$

where $f : \mathbb{R}^p \to \mathbb{R}^d$ and $g : \mathbb{R}^q \to \mathbb{R}^d$ are linear transformations to a joint latent space, such that $d \leq \min\left(p, q\right)$. It can be shown that this equation is maximized through a generalized eigenvector problem.

While CCA has proven to be a very successful algorithm, it does have some limiting features. CCA is only able to accept two input data sets, and these data sets must be of the same cardinality. Moreover, CCA only provides linear subspaces, which may be a limiting factor for data sets with more complex relationships.

2.2 Manifold Alignment

Manifold alignment was introduced as a non-linear alternative to CCA that is able to align multiple data sets that may or may not have corresponding samples between sets [55]. The general manifold alignment framework for two data sets [82] is the following. Given the data sets X and Y of shapes $N_X \times D_X$ and $N_Y \times D_Y$, each row is a sample (or instance) and each column is a feature, and a correspondence matrix $C^{(X,Y)}$ of shape $N_X \times N_Y$, where

$$C_{i,j}^{(X,Y)} = \begin{cases} 1 & : X_i \text{ is in correspondence with } Y_j \\ 0 & : \text{otherwise} \end{cases} \tag{2.1}$$

Manifold alignment calculates the embedded matrices $F^{(X)}$ and $F^{(Y)}$ of shapes $N_X \times d$ and $N_Y \times d$ for $d \leq min(D_X, D_Y)$ that are the embedded representation of X and Y in a shared, low-dimensional space. These embeddings aim to preserve both the intrinsic geometry within each data set and the sample correspondences among the data sets. More specifically, the embeddings minimize the loss function $\mathcal{V}$,

$$\begin{aligned} \mathcal{V}\left(F^{(X)}, F^{(Y)}\right) = &\frac{\mu}{2} \sum_{i=1}^{N_X} \sum_{j=1}^{N_Y} ||F_i^{(X)} - F_j^{(Y)}||_2^2 C_{i,j}^{(X,Y)} \\ &+ \frac{1-\mu}{2} \sum_{i,j=1}^{N_X} ||F_i^{(X)} - F_j^{(X)}||_2^2 W_{i,j}^{(X)} \\ &+ \frac{1-\mu}{2} \sum_{i,j=1}^{N_Y} ||F_i^{(Y)} - F_j^{(Y)}||_2^2 W_{i,j}^{(Y)}, \end{aligned} \tag{2.2}$$

where $\mu \in [0,1]$ is the correspondence tuning parameter, and $W^{(X)}, W^{(Y)}$ are the calculated similarity matrices of shapes $N_X \times N_X$ and $N_Y \times N_Y$, such that

$$W_{i,j}^{(X)} = \begin{cases} k(X_i, X_j) & : X_j \text{ is a neighbor of } X_i \\ 0 & : \text{otherwise} \end{cases} \tag{2.3}$$

for a given kernel function $k(\cdot,\cdot)$. $W_{i,j}^{(Y)}$ is defined in the same fashion. Typically, k is set to be the nearest neighbor set member function or the heat kernel $k(X_i, X_j) = \exp\left(-|X_i - X_j|^2\right)$.

In the loss function of equation (2.2), the first term corresponds to the alignment error between corresponding samples in different data sets. The second and third terms correspond to the local reconstruction error for the data sets X and Y respectively. This equation can be simplified using block matrices by introducing a joint weight matrix W and a joint embedding matrix F, where

$$W = \begin{bmatrix} (1-\mu)W^{(X)} & \mu C^{(X,Y)} \\ \mu C^{(Y,X)} & (1-\mu)W^{(Y)} \end{bmatrix} \tag{2.4}$$

and

$$F = \begin{bmatrix} F^{(X)} \\ F^{(Y)} \end{bmatrix}. \tag{2.5}$$

The loss function $\mathcal{V}$ can be reduced to a matrix trace formulation,

$$\underset{F:F^\top DF=I}{\arg\min}\ \mathcal{V}(F) = \underset{F:F^\top DF=I}{\arg\min}\ tr(F^\top LF), \tag{2.6}$$

where $tr(\cdot)$ is the matrix trace and L is the combinatorial graph Laplacian $L = D - W$, where D is the diagonal matrix of row sums $D(i,i) = \sum_j W(i,j)$ [82].

The constraint $F^\top DF = I$ ensures that the problem is well posed and removes arbitrary scaling factors in the embedding. The d columns of the embedding matrix F

in equation (2.6) are equal to the *d smallest* eigenvectors, the eigenvectors associated with the smallest non-zero eigenvalues, of the Laplacian matrix in the generalized eigenvalue problem $LF = \lambda DF$ [82].

Extensions of Manifold Alignment

In addition to the non-linear manifold alignment technique presented above, there is also a linear formulation that provides a natural map for out-of-sample extensions. This algorithm uses a construction similar to locality preserving projections [42]. A two-step alignment algorithm was described that first uses Procrustes analysis [81]. More recently, an extension of manifold alignment for semi-supervised domain adaptation with categorical labels was detailed [83]. And finally, a global geometry preserving version of manifold alignment was reported [84].

2.3 Mixtures of Manifold Learning

In the literature, mixtures of manifolds of learning are also called *multi-manifold* or *multiple manifold* learning. While the terms may vary, the goal of the field is the same, extending manifold learning to the case of multiple intersecting or nearby manifolds (as shown in Figure 1.1c).

Many mixed manifold methods employ a common approach, using local neighborhood estimates of curvature or shape to discriminate between manifolds. The multiple manifold problem is described in [6], where the authors use local tangent space estimates across all samples to find global trends in the geometry. In [87], the principal angle between local tangent spaces is used to help define neighborhoods on the manifold. In [35], Gaussians are fit over local neighborhoods until the space is covered, then a modified Hellenger distance is used to construct a neighborhood graph over the space of Gaussians where the edges are weighted by Mahalanobis distance.

Other mixed manifold methods have been proposed that do not rely on neighborhood geometry when discriminating manifolds. The problem of multiple intersecting manifolds was first explicitly solved in [72], where an Expectation Maximization style algorithms was used to partition the samples such that within each partition the Euclidean distance within the embedded space best matched an estimated geodesic distance. This approach requires the number of manifolds and their dimensionalities as input, a drawback common to all of the multi-manifold methods listed. In low rank embedding (LRA) [53], a low rank representation is used to construct the manifold adjacency graph. This method is notable because it does not require the number of manifolds as a parameter. However, it its original description, it does not support manifolds of varying dimension.

2.4 Domain Adaptation

Domain adaptation (DA) is a sub-field of transfer learning that uses one or more source data sets to predict a target data set drawn from a different but related distribution. Unsupervised domain adaptation is the subfield of DA that solves problems where no label information is known for either the target or source data sets. These methods are widely studied in the literature [19, 28, 38, 59] and perform well when label information is not available. Unsupervised DA methods are often agnostic to the subsequent task (e.g., classification, regression). While this results in wide applicability, unsupervised DA cannot benefit from label information in cases where it is available.

Supervised and semi-supervised DA are subfields that seek to solve problems where label information is present, if only partially, in the target or source domains. These methods are also widely studied [20,83]; however, prior work has largely assumed that the label information is categorical, as is often the case for classification tasks. In a regression setting with continuous labels, label preprocessing techniques such as bin-

ning or clustering may be used to discretize the label information. These techniques enable the use of existing DA methods, but this work demonstrates that such label manipulation is an imperfect stopgap. Supervised methods that natively handle continuous labels have been proposed [29], but no intra-data set learning happens using the label information and the subsequent learning task is again ignored.

An SDP-based algorithm for domain adaptation for regression is presented in [16], including point-wise learning guarantees. However, it is assumed that the source and target distributions be "reasonably close" and that the feature space be shared. This work is extended in [17] to include new guarantees and a faster algorithm, to address scaling issues present in [16], but the same strong assumptions about the source and target domain similarity pervades. Source and target set bias correction with continuous labels is discussed in [94], but this too requires a homogeneous feature space representation across data sets. Although not explicitly domain adaptation algorithms, in [85, 86], location-scale shifts based on the support of the data are used to transform multiple data sets for regression, but a shared feature space is assumed.

2.5 ChemCam and LIBS

ChemCam is a laser-induced breakdown spectrometer (LIBS) aboard the *Curiosity* rover that analyzes the chemical composition of the rocks, minerals, and soils on the Martian surface. For a period of three years, I was a participating scientist on the ChemCam team of NASA's Mars Science Laboratory. My work with this team has directly motivated my dissertation. This dissertation covers topics related to the instrument, as well as LIBS spectra and spectroscopy in general, so a brief background is provided because it is outside the computer science domain.

To begin, a description of how the instrument records a spectrum from a sample. The LIBS laser pulses the target sample, ablating the surface and creating a plasma. The sample may be up to 7 meters away from the rover, so a telescope is used

to observe the photons emitted as the excited electrons return to their usual valence shells. Using mirrors, the light emitted from the plasma is passed through a diffraction grating to separate the beam into three frequency ranges. The three sub-beams are directed to three charge-coupled devices (CCD), which are sensitive to different frequency ranges. The number of photons that strike the surface of each CCD is recorded to produce the spectrum. Figure 2.1 contains a depiction of the ChemCam instrument sampling a rock on Mars, a closeup of the rock before and after sampling, and the resulting spectra recorded.

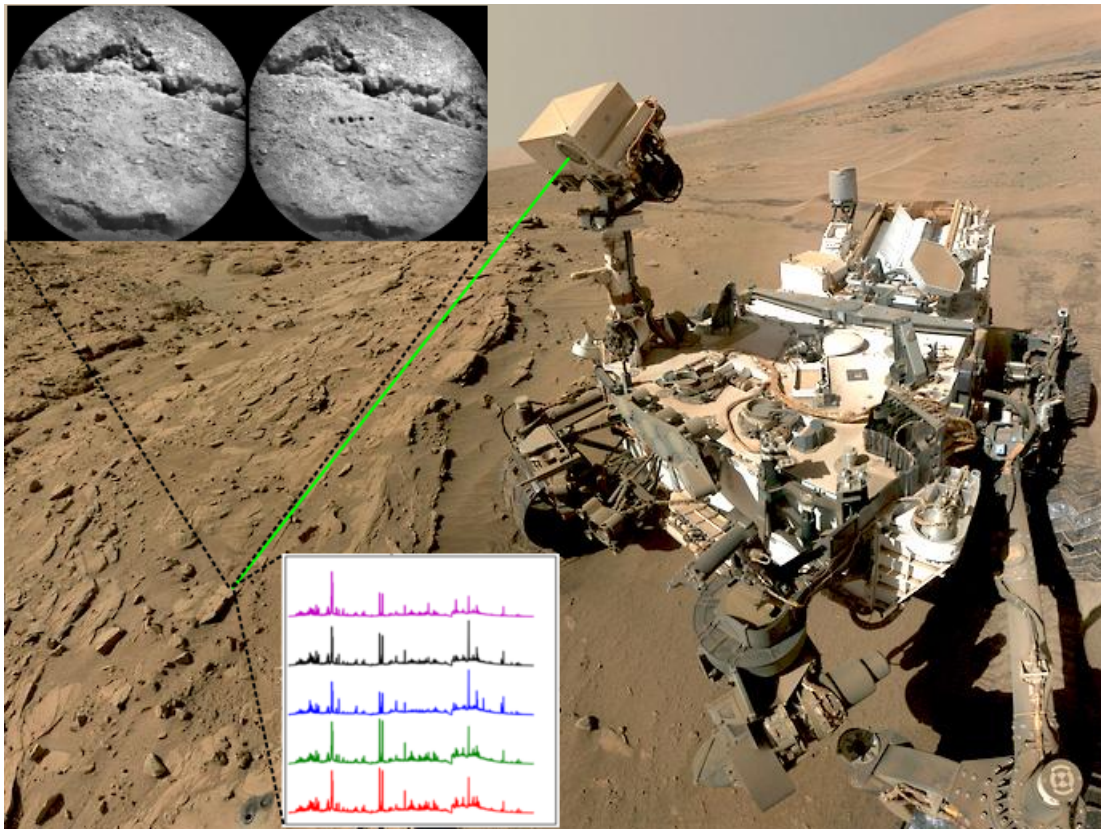


Figure 2.1: *Curiosity* rover on Mars with a simulated ChemCam laser pulse. The photos on the left are of a Martian rock surface before and after laser ablation. The rock was lased 50 times in each of the five locations. In the insert at the bottom are the five mean spectra from each location. Photos courtesy of NASA.

There are four distinguishing characteristics to address with the types of space science data that we will describe. These characteristics are born out of the uniqueness of the challenge inherent in space science and spectroscopic applications. While individually these characteristics are not unique to space science data, the combination presents a unique challenge that has been little studied in the machine learning literature. First, we define space science to be all scientific studies conducted with instruments outside of Earth or its atmosphere, including remote-sensed data of any planet by an orbiting body. When exploring a foreign body in the solar system and beyond, it is often the case that little or no ground-truth data are known about the new system, so these extraterrestrial instruments must be calibrated before launch and subsequently with similar instruments in terrestrial laboratories. In the following section, each of these four characteristics will be explored in detail.

The most distinguishing and consistent characteristic of space science data are that they have many more features than examples. For instance, before sending *Curiosity* to the Martian surface, a small calibration database of pressed rock powders was analyzed using the ChemCam flight model under simulated Mars conditions at Los Alamos National Laboratory [91]. The chemical composition of each powder, expressed as the weight % oxide of elements like SiO_2 and Al_2O_3, of each powder, was known, and a regression model, referred to as a *calibration curve*, was fit on the spectra to predict oxide composition. Each rock powder resulted in a spectrum with 6144 channels each corresponding to a wavelength increment, but only 69 unique rock powders were recorded to form the entire database! To verify the quality of the Earth-based calibration model, a tray of ten sintered rock powders of known composition, referred to as the ChemCam *calibration targets*, was affixed to the rover. So the labeled Martian data are spectra of just ten samples recorded at 6144 channels, though repeatedly analyzed. Models fit on these poorly conditioned problems are unlikely to result in quality predictions. Preprocessing the data using dimensionality reduction

(DR) can remedy this poor conditioning and avoid the demand for more training samples to well fit a model in higher-dimensional space.

The small labeled databases common throughout space science have another unifying similarity: they typically have continuous, or real-valued, label information, and many share a common response surface (e.g., weight % oxide). Consequently, the task associated with the data is generally regression. Supervised DR with continuous labels is certainly not a new problem; the well-known canonical correlation analysis (CCA) algorithm was first described by Hotelling in 1936 [44]. However, most supervised DR algorithms assume the labels are categorical, and so little has been published exploring continuously labeled data that also exhibit these other characteristics of space science data. Furthermore, utilizing the label information is especially important when working with space science data because the feature space is often quite noisy and highly collinear. Precise label information is necessary for analyzing the feature space and determining which channels have a low signal-to-noise ratio.

While labeled space science data often come at a premium, especially *in-situ* ground-truth readings like those from the ChemCam calibration targets, unlabeled data are often abundant. Within the machine learning community, this field is known as *semi-supervised learning*. In this field, information from unlabeled data is gleaned by comparing it with a small set of labeled samples or by analyzing the geometry of the combined feature space. In the case of ChemCam, to date more than 300,000 spectra have been recorded on the Martian surface, where the overwhelming majority are unlabeled (and the remaining spectra are from the ten calibration targets). Fitting a calibration curve using only the 69 pre-flight samples and ignoring the 300,000 unlabeled spectra would be irresponsible of a machine learning practitioner, as would a calibration based only on the ten in-situ targets. As space instrumentation continues to be better engineered, like the Martian rover *Opportunity* that is still gathering

scientific data now at over 47 times its designed lifespan, the semi-supervised nature of the data will only become more pronounced.

The use of overabundant unlabeled samples is not the only means of overcoming the small labeled data sets found in space science. Domain adaptation is another approach that may be used to expand the number of known samples in a training set. This technique is especially well-suited to scientific instrument data, which exhibit distributional variation as a result of differences across instrumentation and environmental conditions. It is often necessary to correct for differences arising from variable experimental geometries (close-up vs. long distance measurements), environmental conditions (e.g., deep sea vs. ambient lab conditions vs. the Martian surface), and analytical parameters such as laser wavelength, power density, and beam size, before the union of disparate data sets becomes advantageous for training. Within the chemistry community, this problem is known as *calibration transfer*. Some of the most successful domain adaptation and calibration transfer methods incorporate DR in their algorithms by selecting only the d-dimensions that are most transferable (by some measure varying with algorithm). By combining domain adaptation and DR into one algorithm, both sub-tasks benefit.

CHAPTER 3

MIXED MANIFOLD ALIGNMENT

One task that has especially benefited from manifold-based approaches is data set alignment, a semi-supervised task in which correspondences are learned between multiple data sets based on intra-set geometry and a provided partial set of pairwise correspondences between the data sets. Manifold alignment is a class of techniques that solves the alignment problem when these data sets are assumed to share a common underlying structure, by embedding each input set into a shared latent manifold space [41, 82].

Manifold alignment was introduced as a semi-supervised, nonlinear extension of canonical correlation analysis (CCA) [44] that aimed to preserve both local geometry and inter-set correspondences [82]. One drawback of existing manifold alignment approaches is their assumption that all data sets are drawn from one or more non-overlapping manifolds. This assumption stems from the use of distance-weighted local neighborhoods for embedding construction, a technique that fails when the data are drawn from an intersecting mixture of manifolds. As data sets begin to use representations that employ thousands or millions of features [54], the assumption of non-mixing manifolds becomes increasingly tenuous.

This mixture of manifolds problem is demonstrated in Figure 1.1: when input manifolds are poorly separated, local neighborhood information is insufficient for recovering the true structure at manifold junctions. Thus, the use of traditional nearest neighbor graph construction algorithms induce incorrect connections at these intersection points, distorting the manifold representation.

This shortcoming of conventional manifold methods has given rise to a number of unsupervised clustering algorithms that attempt to segment input data by identifying the individual manifold components of a mixed manifold structure [72, 88]. One such algorithm is low rank embedding (LRE) [53], which notably avoids the construction of a nearest neighbor graph.

In this chapter, a novel manifold learning algorithm, low rank alignment (LRA), is presented, building on the ideas of manifold alignment and LRE to align data sets drawn from mixtures of manifolds. LRA does not suffer from the sensitive nearest neighbor hyperparameter present in traditional manifold alignment, nor does it require prior knowledge of the number of manifolds, a common requirement for many mixed manifold clustering techniques.

3.1 Low Rank Embedding

Low rank embedding (LRE) is a variation on locally linear embedding (LLE) [65] that uses low rank matrix approximations instead of LLE's nearest neighbor approach to calculate a reconstruction coefficients matrix [53]. LRE is a two part algorithm. Given a data set X, LRE begins by calculating the reconstruction coefficients matrix R by minimizing the loss function

$$\min_R \frac{1}{2}||X - RX||_F^2 + \lambda||R||_*, \tag{3.1}$$

where $\lambda > 0$, $||X||_F = \sqrt{\sum_i \sum_j |x_{i.j}|^2}$ is the Frobenius norm, and $||X||_* = \sum_i \sigma_i(X)$ is the nuclear norm, for singular values σ_i. In [12], it was shown that the nuclear norm is the best convex relaxation of the rank minimization problem, and so the solution RX is a low rank representation of the original data matrix X. To solve equation (3.1), the alternating direction method of multipliers (ADMM) [11] is used.

To apply ADMM, a new variable Z is introduced and equation (3.1) becomes

$$\min_{Z,R} \frac{1}{2}||X - RX||_F^2 + \lambda||Z||_*, \text{ s.t. } R = Z. \tag{3.2}$$

To solve the constrained optimization problem of equation (3.2), the augmented Lagrangian function $\hat{\mathcal{L}}$ is introduced,

$$\begin{aligned}\hat{\mathcal{L}}(Z, R, G) = &\frac{1}{2}||X - RX||_F^2 + \lambda||Z||_* \\ &+ \langle G, R - Z\rangle + \frac{\beta}{2}||R - Z||_F^2,\end{aligned} \tag{3.3}$$

where G is the Lagrange multiplier and $\beta > 0$ is the penalty parameter that controls the convergence of the ADMM algorithm.

The second step of LRE preserves the point-wise linear reconstruction by holding R fixed while minimizing the reconstruction loss in the embedded space,

$$\min_{F^{(X)}} \frac{1}{2}||F^{(X)} - RF^{(X)}||_F^2 \; s.t. \; (F^{(X)})^\top F^{(X)} = I, \tag{3.4}$$

where $F^{(X)}$ is the embedding of X and I is the identity matrix. The constraint $(F^{(X)})^\top F^{(X)} = I$ ensures that it is a well-posed problem. In [67], it was shown that equation (3.4) can be minimized by calculating the d *smallest* non-zero eigenvectors of the Gram matrix $(I - R)^\top (I - R)$.

3.2 Low Rank Alignment

Low rank alignment (LRA) is a novel algorithm for the manifold alignment task that uses a variant of LRE to embed the data sets to a joint manifold space, unlike previous alignment methods that have been based on Laplacian eigenmaps [5, 81, 82] and Isomap [76, 84]. These methods rely on nearest neighbor graph construction

algorithms, and are thus prone to creating spurious inter-manifold connections when mixtures of manifolds are present. These so-called *short-circuit connections* are most commonly found at junction points between manifolds. In contrast, LRA is able to avoid this problem, successfully aligning data sets drawn from a mixture of manifolds. Figure 3.1 shows an example of this phenomena using a noisy dollar sign data set.

LRA differs from other manifold alignment algorithms in several key aspects. While some previous algorithms embed data using exclusively the eigenvectors of the graph Laplacian to preserve both inter-set correspondences and intra-set local geometry, LRA uses the eigenvectors of the sum of the Laplacian and the Gram matrix of low rank representations to preserve the inter-set correspondences and the intra-set *local linearity*. Moreover, previous manifold alignment algorithms require a reliable measure of similarity between nearest neighbor samples, whereas LRA relies on the linear weights used in sample reconstruction. Lastly, because LRA uses the global property of rank to calculate its reconstruction matrix, it can better discern the global structure of mixing manifolds [53].

We now describe the low rank alignment algorithm for two data sets. It begins with the same setup as manifold alignment: two data sets X and Y are given, along with the correspondence matrix $C^{(X,Y)}$ describing inter-set correspondences (see equation 2.1). The goal of LRA is to calculate a set of embeddings $F^{(X)}$ and $F^{(Y)}$ to a joint, low-dimensional manifold subspace that best preserves both inter-set correspondences and intra-set geometries.

In the first step of LRA, the reconstruction weight matrices $R^{(X)} \in \mathbb{R}^{N_X \times N_X}$ and $R^{(Y)} \in \mathbb{R}^{N_Y \times N_Y}$ are calculated individually according to equation (3.1). In this step, the low rank constraint defines a barycentric coordinate for each sample that preserves locally linear relationships between samples. In [27], it is shown that the low rank representation problem in equation (3.1) can be solved in closed form. This avoids the iterative ADMM calculation found in the original LRE algorithm.

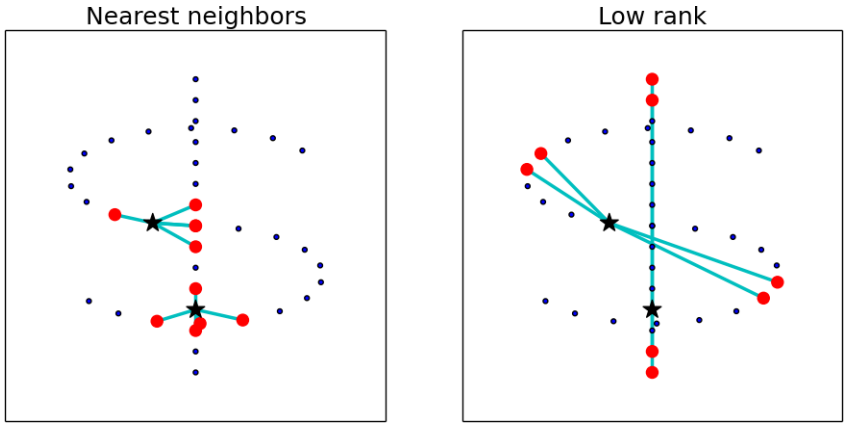


Figure 3.1: Manifold construction on synthetic data. For each star, the large points are the four neighbors used to define the manifold. The lines match the stars with their neighbors. Notice that traditional nearest neighbor construction (on the left) has *short-circuits* incorrectly connecting the two manifolds, whereas the low rank construction (on the right) selects points that correctly differentiate the mixed manifolds.

We begin by decomposing X using singular value decomposition (SVD), $X = USV^\top$. Next, the columns of V and S are partitioned into $V = [V_1 V_2]$ and $S = [S_1 S_2]$ according to the sets

$$I_1 = \{i : s_i > 1 \ \ \forall s_i \in S\} \text{ and } I_2 = \{i : s_i \leq 1 \ \ \forall s_i \in S\}.$$

Then the reconstruction matrix $R^{(X)}$ is calculated as

$$R^{(X)} = V_1(I - S_1^{-2})V_1^\top. \tag{3.5}$$

$R^{(X)}, R^{(Y)}$ are calculated independently and so may be computed in parallel to reduce compute time. We next define the block matrices $R, C \in \mathbb{R}^{N \times N}$ as

$$R = \begin{bmatrix} R^{(X)} & 0 \\ 0 & R^{(Y)} \end{bmatrix} \text{ and } C = \begin{bmatrix} 0 & C^{(X,Y)} \\ C^{(Y,X)} & 0 \end{bmatrix} \tag{3.6}$$

and $F \in \mathbb{R}^{N \times d}$ as

$$F = \begin{bmatrix} F^{(X)} \\ F^{(Y)} \end{bmatrix}. \tag{3.7}$$

The second step of LRA is to calculate the embedding F of X, Y by minimizing the loss function

$$\mathcal{Z}(F) = (1-\mu)\left\|F - RF\right\|_F^2 + \mu \sum_{i,j=1}^{N} ||F_i - F_j||^2 C_{i,j}, \tag{3.8}$$

where $\mu \in [0,1]$ is the hyperparameter that controls the importance of inter-set correspondences. The first term of the sum in equation (3.8) accounts for the local geometry within each data set, and the second term accounts for the correspondences between sets. We can then reduce this loss function to a sum of matrix traces:

$$\begin{aligned}
\mathcal{Z}(F) &= (1-\mu) tr((F-RF)^\top (F-RF)) \\
&\quad + \mu \sum_{k=1}^{d} \sum_{i,j=1}^{N} ||F_{i,k} - F_{j,k}||_2^2 C_{i,j} \\
&= (1-\mu) tr\left(((I-R)F)^\top (I-R)F\right) \\
&\quad + 2\mu \sum_{k=1}^{d} F_{\cdot,k}^\top L F_{\cdot,k} \\
&= (1-\mu) tr(F^\top (I-R)^\top (I-R)F) \\
&\quad + 2\mu \; tr(F^\top L F).
\end{aligned} \tag{3.9}$$

As with LLE and LRE, we introduce the constraint $F^\top F = I$ to ensure that the minimization of the loss function $\mathcal{Z}$ is a well-posed problem. Thus, we have

$$\underset{F:F^\top F=I}{\arg\min}\, \mathcal{Z} = \underset{F:F^\top F=I}{\arg\min} (1-\mu) tr(F^\top M F) + 2\mu \; tr(F^\top L F), \tag{3.10}$$

where $M = (I - R)^\top (I - R)$. To construct a loss function from equation (3.10), we take the right hand side and introduce the Lagrange multiplier Λ,

$$\begin{aligned}\mathcal{L}(F, \Lambda) &= (1-\mu) tr(F^\top M F) + 2\mu \; tr(F^\top L F) \\ &\quad + \langle \Lambda, F^\top F - I \rangle. \end{aligned} \tag{3.11}$$

To minimize equation (3.11), the roots of its partial derivatives must be found,

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial F} &= 2(1-\mu) M F + 4\mu L F - 2\Lambda F = 0 \\ \frac{\partial \mathcal{L}}{\partial \Lambda} &= F^\top F - I = 0. \end{aligned} \tag{3.12}$$

From this system of equations, results the matrix eigenvalue problem

$$\left((1-\mu) M + 2\mu L\right) F = \Lambda F \quad \text{and} \quad F^\top F = I. \tag{3.13}$$

Therefore, to solve equation (3.10), calculate the d *smallest* non-zero eigenvectors of the matrix

$$(1-\mu) M + 2\mu L. \tag{3.14}$$

This eigenvector problem can be solved efficiently because the matrix $M + L$ is guaranteed to be symmetric, positive semidefinite (PSD), and sparse. These properties arise from the construction,

$$\begin{aligned} M + L = &\begin{bmatrix} \left(I - R^{(X)}\right)^2 & 0 \\ 0 & \left(I - R^{(Y)}\right)^2 \end{bmatrix} \\ &+ \begin{bmatrix} D^X & -C^{(X,Y)} \\ \left(-C^{(X,Y)}\right)^\top & D^Y \end{bmatrix}, \end{aligned} \tag{3.15}$$

where by construction $D = \begin{bmatrix} D^X & 0 \\ 0 & D^Y \end{bmatrix}$ is a PSD diagonal matrix and $C^{(X,Y)}$ is a sparse matrix.

The time complexity of LRA is dominated by two operations: the full singular value decomposition necessary for step 1 and the sparse eigenvector decomposition in step 2. The runtime of the SVD is cubic $\mathcal{O}\left(\max(N_X, X_y)^3\right)$ proportional to the number of samples in the set. Calculating the reconstruction matrix R in step 1 of LRA is naturally parallelizable by data set, so the cost of the SVD is limited to the individual data set size. The eigenvector decomposition also runs in cubic, but it is proportional to the total number of samples $\mathcal{O}\left((N_X + N_Y)^3\right)$. This calculation is only performed once during the algorithm. Because the matrix is symmetric, it can be first converted into a tridiagonal Hessenberg matrix using the Lanczos algorithm, and then the QR algorithm can be used to find the eigenvectors [39].

Algorithm 1: Low Rank Alignment

Input: data matrices X, Y, embedding dimension d,
correspondence matrix $C^{(X,Y)}$ and weight μ.
Output: embeddings matrix F.
Step 0: Column normalize X & Y *(optional but recommended if X and Y differ largely in scale).*
Step 1: Compute the reconstruction coefficient matrices $R^{(X)}, R^{(Y)}$:
$USV^\top = \text{SVD}(X)$
$R^{(X)} = V_1(I - S_1^{-2})V_1^\top$
$\hat{U}\hat{S}\hat{V}^\top = \text{SVD}(Y)$
$R^{(Y)} = \hat{V}_1(I - \hat{S}_1^{-2})\hat{V}_1^\top$
Step 2: Set F equal to the d *smallest* eigenvectors of the matrix in equation (3.14).

3.3 Kernelized LRA

In the first step of low rank alignment, a reconstruction matrix is calculated for each data set to capture the underlying subspaces. In practice, for high-dimensional data this linear reconstruction will suffice to describe the underlying space, but some

data drawn from complex mixtures of manifolds require a non-linear reconstruction. To this end, a kernelized formulation of low rank alignment is derived here. This requires modifying the first step of the LRA algorithm, where the reconstruction matrices are calculated for each data set.

Given a data set $X \in \mathcal{X}$, let ϕ be a feature map from $\mathcal{X}$ to a possibly infinite dimensional inner product space $\mathcal{V}$. On $\mathcal{V}$, the inner product can be described by a kernel function $k(x, x') = \langle \phi(x), \phi(x') \rangle_{\mathcal{V}}$, and the *kernel trick* can be used. Rewriting equation 3.1 with this feature map yields

$$\min_R \frac{\omega}{2} \|\Phi - R\Phi\|_F^2 + \|R\|_* \tag{3.16}$$

where $\Phi = [\phi(X_1), \ldots, \phi(X_N)]$.

Theorem 1. *Let K be the kernel matrix $K = \Phi\Phi^\top$ and $U, \Sigma, V^\top = SVD(K)$. The optimal solution $\hat{R}$ to equation 3.16 is given by the formula*

$$\hat{R} = U_1 \left(I - \frac{1}{\omega} \Sigma_1^{-1} \right) U_1^\top, \tag{3.17}$$

where Σ_1 is the diagonal matrix of singular values greater than ω^{-2} and U_1 are the columns of U associated with Σ_1.

Before proving this theorem, some basic facts must be covered. First, in addition to defining Σ_1, define Σ_2 as the diagonal matrix of singular values less than or equal to ω^{-2} and U_2 as the columns of U associated with Σ_2, i.e., $U = [U_1 U_2], V = [V_1 V_2]$. Because these subset matrices have orthogonal columns, they have a left inverse relationship $U_1^\top U_1 = I$ and $V_1^\top V_1 = I$ (and likewise for U_2 and V_2), and the property $U_1 U_2^\top = 0$ and $U_2 U_1^\top = 0$. Furthermore, from this construction comes the relationship

$$K = U_1 \Sigma_1 V_1^\top + U_2 \Sigma_2 V_2^\top. \tag{3.18}$$

From these two facts, it can be deduced that

$$U_1U_1^\top + U_2U_2^\top = V_1V_1^\top + V_2V_2^\top = I. \tag{3.19}$$

The proof for theorem 1 is now provided.

Proof. Expanding equation 3.16 yields

$$\min_R \frac{\omega}{2}\left(\text{tr}\left(\Phi\Phi^\top R^\top R\right) - 2\text{tr}\left(\Phi\Phi^\top R\right)\right) + \|R\|_*. \tag{3.20}$$

Now that the equation has been written in terms of inner products $\Phi\Phi^\top$, the kernel trick can be applied to 3.20 to yield

$$\min_R \frac{\omega}{2}\left(\text{tr}\left(KR^\top R\right) - 2\text{tr}\left(KR\right)\right) + \|R\|_*. \tag{3.21}$$

Completing the square in this expansion yields the equivalent

$$\min_R \frac{\omega}{2}\|K^{\frac{1}{2}} - RK^{\frac{1}{2}}\|_F^2 + \|R\|_*. \tag{3.22}$$

To optimize equation 3.22, we calculate its differential as

$$\omega K\left(R - I\right) + \partial_R\|R\|_*, \tag{3.23}$$

where $\partial_R\|R\|_*$ is the subdifferential of R. It suffices to show that equation 3.23 evaluated at $\hat{R}$ contains zero matrix.

Given the compact SVD of $R = U_R \Sigma_R V_R^\top$, the subdifferential of the nuclear norm of a matrix is

$$\partial_R \|R\|_* = \{U_R V_R^\top + W : U_R^\top W = 0, W V_R = 0, \|W\| \leq 1\}, \tag{3.24}$$

[89]. Substituting this in equation 3.23 for $U_R = V_R = U_1$ yields

$$\omega K (R - I) + U_1 U_1^\top + W = 0. \tag{3.25}$$

From equation 3.18 and because K is a symmetric matrix, it follows that

$$K = U_1 \Sigma_1 U_1^\top + U_2 \Sigma_2 U_2^\top \tag{3.26}$$

Plugging in the solution $\hat{R}$ and using equation 3.19 gives

$$I - \hat{R} = I - U_1 \left(I - \frac{1}{\omega} \Sigma_1^{-1} \right) U_1^\top = \frac{1}{\omega} U_1 \Sigma_1^{-1} U_1^\top + U_2 U_2^\top. \tag{3.27}$$

Combining equations 3.26 and 3.27 yields

$$K (I - R) = \left(U_1 \Sigma_1 U_1^\top + U_2 \Sigma_2 U_2^\top \right) \left(\frac{1}{\omega} U_1 \Sigma_1^{-1} U_1^\top + U_2 U_2^\top \right) \tag{3.28}$$

$$= \frac{1}{\omega} U_1 \Sigma_1 U_1^\top U_1 \Sigma_1^{-1} U_1^\top + U_2 \Sigma_2 U_2^\top U_2 U_2^\top \tag{3.29}$$

$$= \frac{1}{\omega} U_1 U_1^\top + U_2 \Sigma_2 U_2^\top. \tag{3.30}$$

Substituting this back into equation 3.25 returns

$$U_1 U_1^\top + W - \omega \left(\frac{1}{\omega} U_1 U_1^\top + U_2 \Sigma_2 U_2^\top \right) = 0, \tag{3.31}$$

which yields the result $W = \omega U_2 \Sigma_2 U_2^\top$. Plugging this back into equation 3.24 confirms that

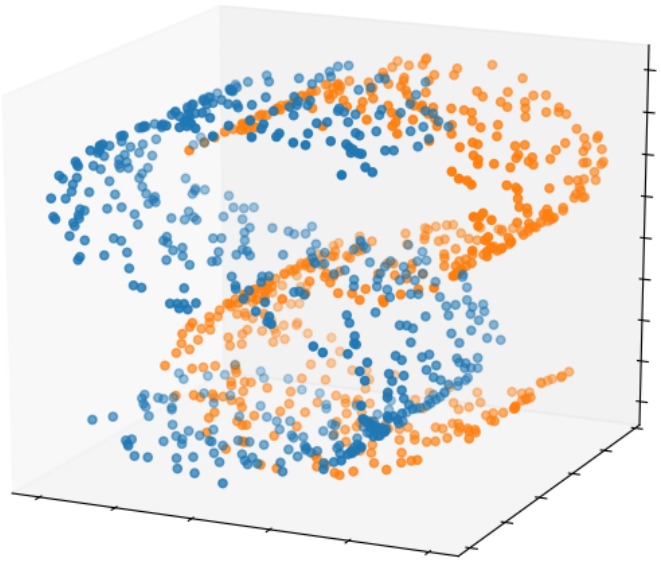
$$\omega U_1^\top U_2 \Sigma_2 U_2^\top = 0, \tag{3.32}$$

$$\omega U_2 \Sigma_2 U_2^\top U_1 = 0, \tag{3.33}$$

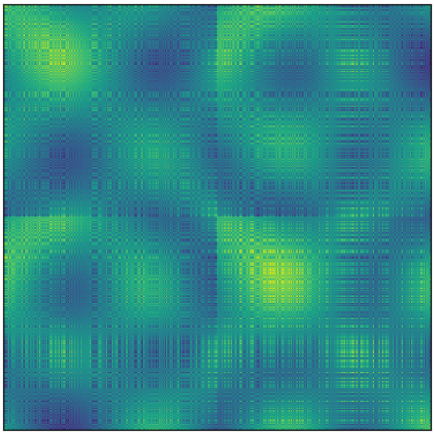
$$\|U_2 \Sigma_2 U_2^\top\| = \|\Sigma_2\| \leq 1/\omega. \tag{3.34}$$

□

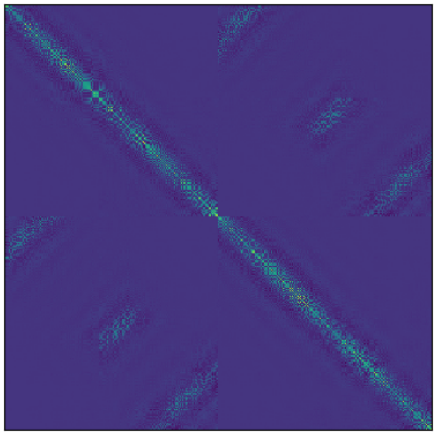
With this kernel formulation, LRA is able to parse and align non-linear manifold subspaces. To show the difference in reconstruction between LRA and kernel LRA, the reconstruction matrices from the two variants are compared using a small synthetic data set. These data are points sampled from two intermixed S-curve manifolds, colored blue and orange in Figure 3.2 (a), and are ordered by manifold. An ideal reconstruction matrix should have all zero entries in the off-diagonal blocks and any non-zero entries in the diagonal blocks. The two manifolds are sampled an equal number of times, so all blocks of the matrix should be of equal size. As shown in Figure 3.2 (b), standard LRA confuses the two non-linear manifolds, with many low-value non-zero entries in the off-diagonal blocks. In contrast, kernel LRA has only trace noise values in the off-diagonal entries, as shown in Figure 3.2 (c), and it uses more of a neighborhood-based reconstruction.



(a) Mixture of s-curves in ambient space.



(b) Reconstruction matrix from low rank alignment (LRA).



(c) Reconstruction matrix from kernel LRA.

Figure 3.2: When embedding highly non-linear and entangled manifolds, like this mixture of s-curves (a), kernel LRA (c) proves advantageous over standard LRA (b), as is demonstrated by the reconstruction matrices. LRA (b) conflates the two curves as seen in the non-zero off-diagonal blocks, whereas KLRA (c) has only trace noise in those entries.

3.4 Clustering with LRA

In addition to aligning and embedding high-dimensional data, LRA is also very effective at clustering data. LRA has two key advantages over other clustering methods. (1) It is able to distinguish between mixtures of manifolds, even at challenging manifold junction areas, and so it is able to separate clustered data by manifolds. Other clustering methods that rely only on a single manifold assumption, Euclidean distance, or probability measures may fail. (2) It is able to cluster across multiple heterogeneous data sets that do not share a common feature representation. LRA accomplishes this by simultaneously aligning multiple heterogeneous data sets and clustering them. Furthermore, by aligning the disparate data sets first, LRA can transfer knowledge between the sets, whereby it is able to cluster multiple data sets more accurately than it could cluster any of the single data sets alone.

For example, when classifying stars and star systems, it is common practice to observe the object with multiple instruments that record at differing wavelengths and resolutions. These objects can be used as correspondences between instruments to help in the alignment of the discordant data sources. It has been shown that the high-dimensional spectra collected from these objects by the sensors lie on a low-dimensional manifold or mixture of manifolds [57, 78]. By aligning the data sensors while clustering, LRA is able to transfer knowledge between instruments to better analyze the astronomical objects.

To cluster heterogeneous data sets, LRA first calculates the data reconstruction matrix $R = diag\left(R^{(1)}, \ldots, R^{(k)}\right) \in \mathbb{R}^{N \times N}$ by solving equation 3.1. After calculating R, the matrix is post-processed to encourage samples on the same manifold to be embedded in similar regions. To do this, an auxiliary version of R is calculated $\hat{R} \in \mathbb{R}^{N \times N}$ as

$$\hat{R}_{i,j} = |R_{i,j}| \left(\sum_{j=1}^{N} |R_{i,j}| \right)^{-1}. \tag{3.35}$$

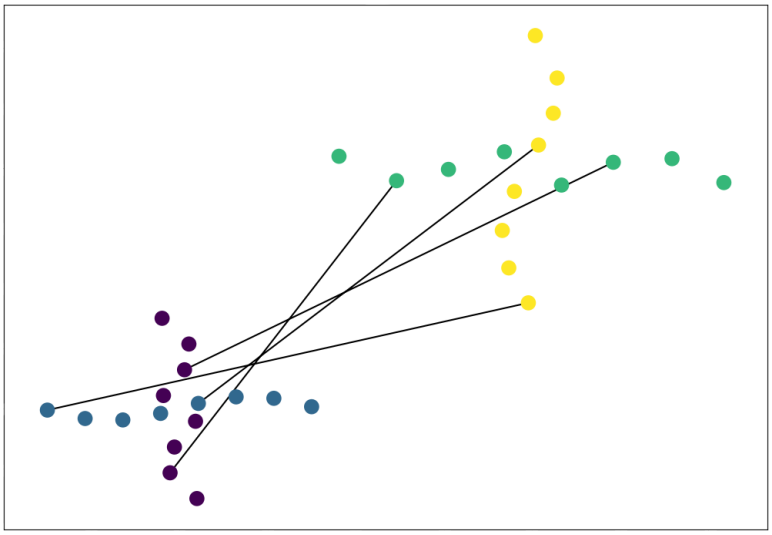
Thus, the zero blocks in R are preserved in $\hat{R}$, but the non-zero block diagonals are normalized such that the rows are convex combinations, i.e., all rows are non-negative entries that sum to 1. This is a simple transformation that greatly improves the clusterings produced by LRA. The transformation can be interpreted as a two-part procedure, replacing signed values with magnitudes and scaling entries. Both of these actions encourage samples from the same manifold to be embedded in the same small neighborhood. By using magnitudes instead of signed values, all points drawn from a single manifold are forced to embed in the same region. Scaling the rows forces the manifold groupings to be more densely packed. After calculating $\hat{R}$, the clustering embedding F is calculated in the same way as standard LRA, by solving the eigenvalue problem of equation 3.14, where M is replaced by

$$\hat{M} = (I - \hat{R})^\top (I - \hat{R}). \tag{3.36}$$

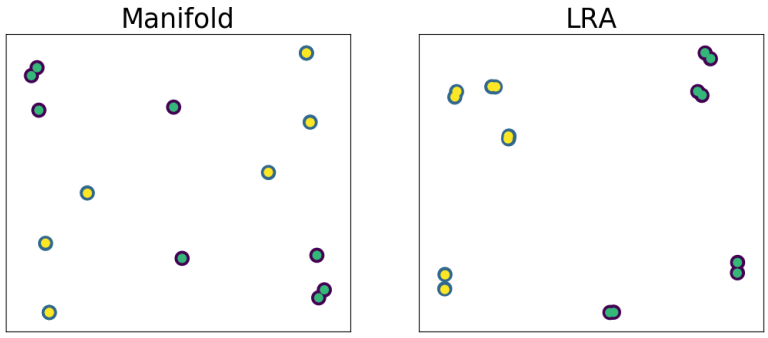
Lastly, the k-means clustering algorithm is applied to F in the low-dimensional shared space. A step by step listing of the algorithm is presented in algorithm 2.

Algorithm 2: LRA-Cluster

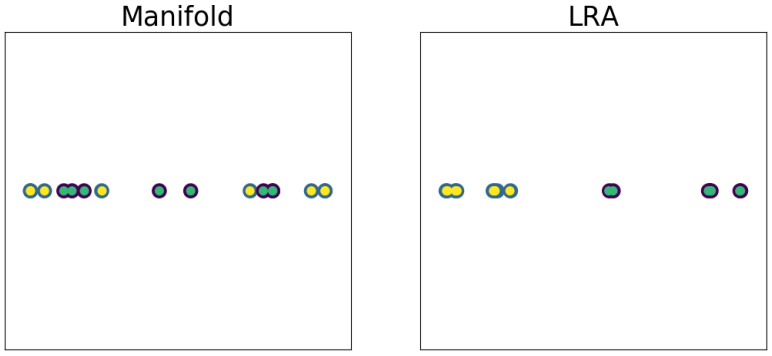
1. Calculate the reconstruction matrix $R^{(i)}$ for each data set $X_i \in X_1 \dots X_k$ to form R.
2. Post-process the reconstruction matrix R to form $\hat{R}$ according to equation 3.35.
3. Calculate the embedding for each sample F by solving the eigenvalue problem $(1 - \mu)\hat{M}F + \mu CF$.
4. Use k-means to cluster the aligned embedding F.



(a) Two mixture of manifold data sets in a shared 2-D ambient space, with example correspondences indicated by black lines.



(b) The 2-D embedding from manifold alignment and low rank alignment.



(c) The 1-D embedding from manifold alignment and low rank alignment.

Figure 3.3: A comparison of 1-D and 2-D embeddings for cluster analysis of two data sets, where each is a mixture of two sinusoidal manifolds. The ambient dimension of the data is two, but the intrinsic dimension of all manifolds is one. LRA successfully disentangles the mixtures of manifolds, aligns the data sets, and embeds the samples for cluster analysis. Manifold alignment successfully aligns the data sets, but fails to separate the mixture of manifolds, so cluster analysis remains difficult with this embedding.

To demonstrate LRA for clustering, it was first applied to a simple synthetic data set in Figure 3.3. In this example, there were two data sets where each set contains two mixed manifolds, where one manifold was sampled from a 1-D manifold and the second was a parameterized sine curve. The second data set used the same parameterization as the first data set, so that all points between data sets would be in correspondence. Moreover, the manifolds in the second data set were linearly transformed by scaling, translation, and rotation and non-linearly transformed by adding noise to the parameterization and varying the period of the sine function. To show how traditional manifold methods fare at aligning and clustering the data, LRA was compared to manifold alignment (MA). In this case, MA tangles the mixtures of manifolds together, so while it successfully aligns the data sets, it cannot pull apart the mixed manifolds. In contrast, LRA for clustering was able to completely and flawlessly separate the mixtures of manifolds while aligning the data sets. Furthermore, the preprocessed auxiliary reconstruction matrix $\hat{R}$ pushed the clusters closer together so that k-means could handily and perfectly divide the two clusters. However, the MA embedding was a much greater challenge for k-means, where a clean partitioning was not possible.

3.5 Actively Learning Correspondences

Typically, machine learning algorithms are passed a set of labeled or unlabeled data, in batch or on-line, and the algorithm learns from the data it passively receives. *Active learning* is a sub-field of machine learning that studies algorithms that can actively request (labeled) examples from the user. In certain domains, it is often expensive and laborious, and sometimes impossible, to collect labeled data. When modeling physical systems, like fluid dynamics or particle interactions, labels can often only be attained after running intensive computer simulations. When working with privacy-constrained data, the cost to obtain permission may be great. Chal-

lenges may be more physical, as when calibrating scientific instrumentation. To build an instrument-specific calibration regression curve appropriate for a certain class of samples, *standards*, samples of known analyte value, must be collected or created and then sampled by the instrument. In cases like these, it is highly advantageous for a learning algorithm to select which samples would be most helpful. Rather than randomly receiving more labels, like a passive online learner, active learners can learn faster by requesting labels for high-value samples.

The general active learning setup is the following: given a set of unlabeled or partially labeled data, the learning algorithm, whose goal is to calculate a map between samples and labels, must request labels for unlabeled examples it has seen or has generated. Not all labeled samples will be equally helpful to the learner, whether it be because of noise, geometry, etc, so the goal of the active learner is to identify these samples. In the task of multi-data set alignment, correspondences between sets can be difficult to acquire. In the previous instrumentation example, acquiring correspondences to aid alignment between multiple instruments would require physically sampling the same standard on at least two of the instruments. In the case of cross-lingual document retrieval, attaining new correspondences requires an expert to translate text between languages. In the simplest case, acquiring a correspondence means requesting a label, but for a full correspondence with $n-1$ labels for n-data sets, actively learning correspondences is especially beneficial.

LRA is naturally amenable to actively learning correspondences. In practice, LRA primarily uses a smaller subset of the data during the reconstruction step, and it is a simple matter to identify this subset. To modify algorithm 1, mean center the data set samples, then a step can be added between 1 and 2 that identifies key samples in each data set to select as correspondences. After calculating a reconstruction matrix $R^{(k)} \in \mathbb{R}^{N_k \times N_k}$ for the centered data sets $X_1, \ldots, X_n$, the active learning importance score vector $\tau\left(R^{(k)}\right) \in \mathbb{R}^{N_k}$ can be calculated as

$$\tau\left(R^{(k)}\right) = \sum_{i \neq j}^{N_k} |R^{(k)}_{i,j}| + |R^{(k)}_{j,i}|. \tag{3.37}$$

The importance score τ calculates the total magnitude by which each sample is used to reconstruct other samples. Intuitively, the importance score of a sample is large when it is frequently used to reconstruct other samples and low otherwise. In some cases where the magitude of the samples in a data set vary greatly, equation 3.37 can be modified to incorporate the norm $\|x_i\|$ as a multiplicative scaling factor to each sample score.

Once τ has been calculated for all data sets, correspondences can be selected from each. If an algorithm is allotted c correspondences, then one scheme to pick samples is to greedily select those c/k from each data set with the largest τ score. This process continues until all k data sets are processed and c correspondences are requested. If there is prior knowledge about the data sets being aligned, different selection scheme may be more appropriate. For example, if one data set is known to be noisier than others, then fewer samples can be selected from this noisy data set when choosing correspondences. An overview of the steps is listed in algorithm 3.

Algorithm 3: Active Learning Correspondences for LRA

Input : R, a k block diagonal reconstruction matrix,
c, number of correspondences.
Output: $Indices$, a k-dimensional array containing indices of selected correspondences.

```
begin
    Indices[k] ← Null
    for R^(i) : i ∈ (1, ..., k) do
        T ← τ(R^(i))    (from equation 3.37)
        for j : (1, ..., c/k) do
            I ← argmax(T)
            Indices[k].add(I)
            T[I] = 0
        end
    end
end
```

The runtime for the active learning step is minimal because the reconstruction matrix R is calculated in the first step of LRA. For each of the n_k samples in the k data sets, the magnitude of its reconstruction matrix $R^{(k)}$ must be calculated, which can be accomplished in $\mathcal{O}(k \cdot n_k^2)$. This results in k vectors of length n_k. The maximum element of a vector can be found in linear time $\mathcal{O}(n_k)$, and this must be done c/k times for each of the k data sets costing $\mathcal{O}(c \cdot n_k)$. The total runtime is the quadratic $\mathcal{O}(c \cdot n_k)$. This results in a runtime of $\mathcal{O}(k \cdot n_k^2 + c \cdot n_k)$. If c is large (i.e, close to n_k in value), it may be more efficient to first presort the vectors magnitudes and select in bulk. This active learning scheme is also naturally parallelizable by data sets. Furthermore, for very large n_k, the rows of R can be batched and the sums performed in parallel.

3.6 Experimental Results

To evaluate the effectiveness of LRA and kernel LRA (KLRA), experiments were performed on two very different types of real-world data, spectroscopic data sets and cross-lingual documents. For comparison, three state of the art alignment techniques were implemented: (instance-level/non-linear) manifold alignment [82], affine matching alignment [51], and Procrustes alignment [81]. All of the methods evaluated align data sets by embedding the sets into a shared low-dimensional space. Affine matching and Procrustes alignment can only align two data sets at a time, and while LRA and manifold alignment do not suffer this limitation, we chose to limit our experimentation to alignment problems involving pairs of data sets.

All experiments were implemented in Python by the author, with help from the machine learning library Scikit-learn [61]. An implementation of LRA is available for download on the author's website. [1]

[1] `https://github.com/all-umass/lowrank_alignment`

3.6.1 Calibration Transfer

The task of these experiments is calibration transfer (CT). CT is a transfer learning problem well-studied in chemometrics [10, 30, 64, 93], but largely unknown to the machine learning community. The general setup of the problem is the following. The spectra of a set of samples (e.g., rock powders) are recorded on different instruments or on the same instrument under varying conditions. The goal is to find a mapping or an alignment between the two (or more) sets of spectra. Frequently in all types of spectroscopic studies there is a need to ensure that possible differences in environmental or experimental conditions are mitigated or negated, allowing data from multiple instruments to be compared. CT provides an excellent solution to the task of reconciling data in inter- and intra-lab comparisons on Earth and in extraterrestrial applications.

Aligning Processed and Raw Spectra

The first data set was a suite of laser-induced breakdown spectra (LIBS) acquired from 100 different geological samples under Mars-like atmospheric conditions at Mount Holyoke College. LIBS instruments are spectrometers composed of a high energy laser that pulses a sample to create plasma, which is observed by a charge-coupled device (CCD) that records the energy emitted. This data set was created in support of the Mars Science Laboratory mission for the ChemCam instrument, the LIBS spectrometer on the rover *Curiosity* [25, 77].

The spectra are provided in two different formats, a raw unprocessed format recorded directly from the instrument and a processed format that has been cleaned using the standard ChemCam routine [91]. Briefly, the preprocessing routine includes: a non-linear transformation to adjust for the CCD's sensitivity to particular wavelength regions, iterative wavelet-based noise removal for both background radiation and *Bremsstrahlung* (the interference from colliding particles in the plasma), shot

averaging over a series of pulse integrations, and wavelength-region-specific feature normalization. The raw spectra have 6609 channels in their feature representation, while the processed spectra have been transformed to a 6144 channel representation. In Figure 3.4, example raw and processed spectra from the data sets are shown.

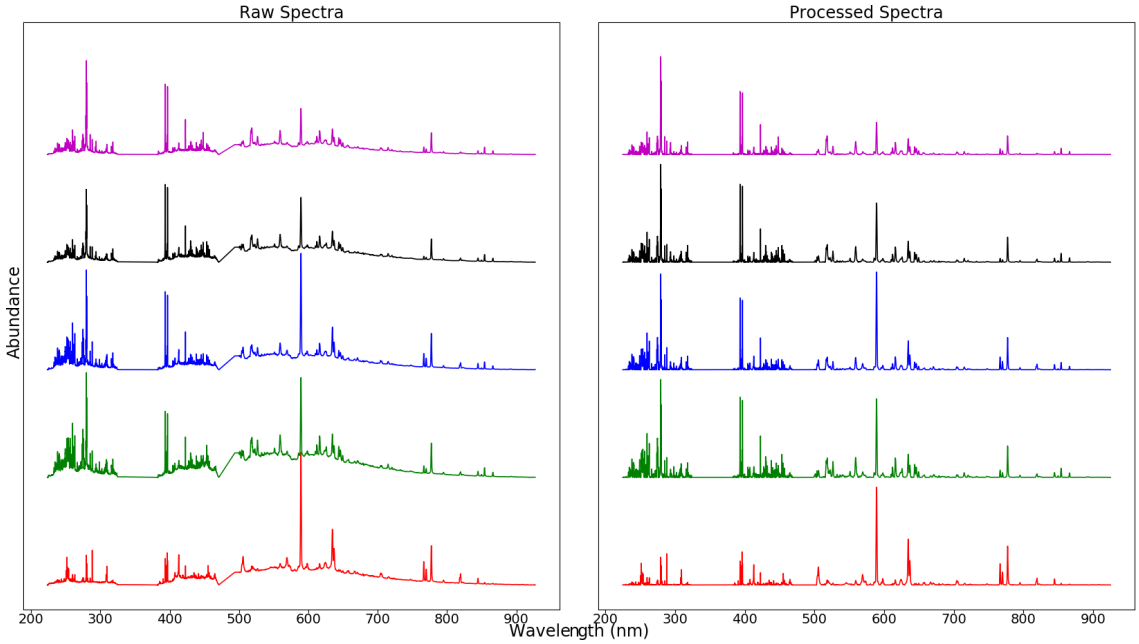


Figure 3.4: Five mineral spectra selected at random from the 100 sample LIBS data sets. The left hand side shows the raw unprocessed spectra, and the right hand side shows the corresponding spectra after a series of processing steps.

The task of the experiment was to align the set of raw spectra with the set of processed spectra. 5-fold cross validation was used to evaluate the competing methods. In each iteration, correspondences were provided for 80 spectra while the other 20 spectra were used for evaluation. A raw test spectrum was considered correctly aligned if the corresponding processed spectrum was its nearest neighbor in the embedded space. The results of the experiment are plotted in Figure 3.5.

LRA was the top performing model of those evaluated, with an accuracy of 84% at embedding dimension $d = 16$. Kernel LRA (KLRA), using an RBF kernel with $\gamma = 0.1$, performed nearly as well with an accuracy of 78% at $d = 17$. The other three

competing models performed about half as well as the LRA methods, with Procrustes alignment and affine matching both getting 36% accuracy.

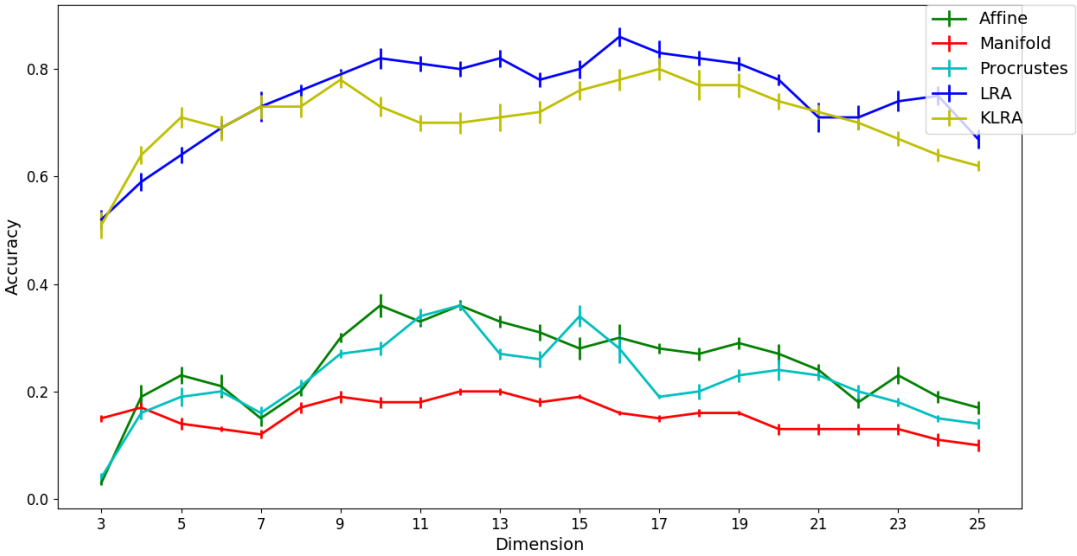


Figure 3.5: Cross validation results of 100 sample raw/processed LIBS spectra alignment experiment, including bars indicating the standard error of the mean.

High and Low Laser Power Spectra

The next CT experiment used a 159 sample LIBS data set of crushed mineral mixtures recorded at Mount Holyoke College. The samples were recorded on the same instrument under a low and high laser power setting, 3% and 5% power respectively, as seen in Figure 3.6. Like the previous experiment, the spectra were processed according to [91]. The resultant spectra were 6144-dimensional real-valued vectors, where each feature corresponded to the response of a particular wavelength channel between 225-925 nm.

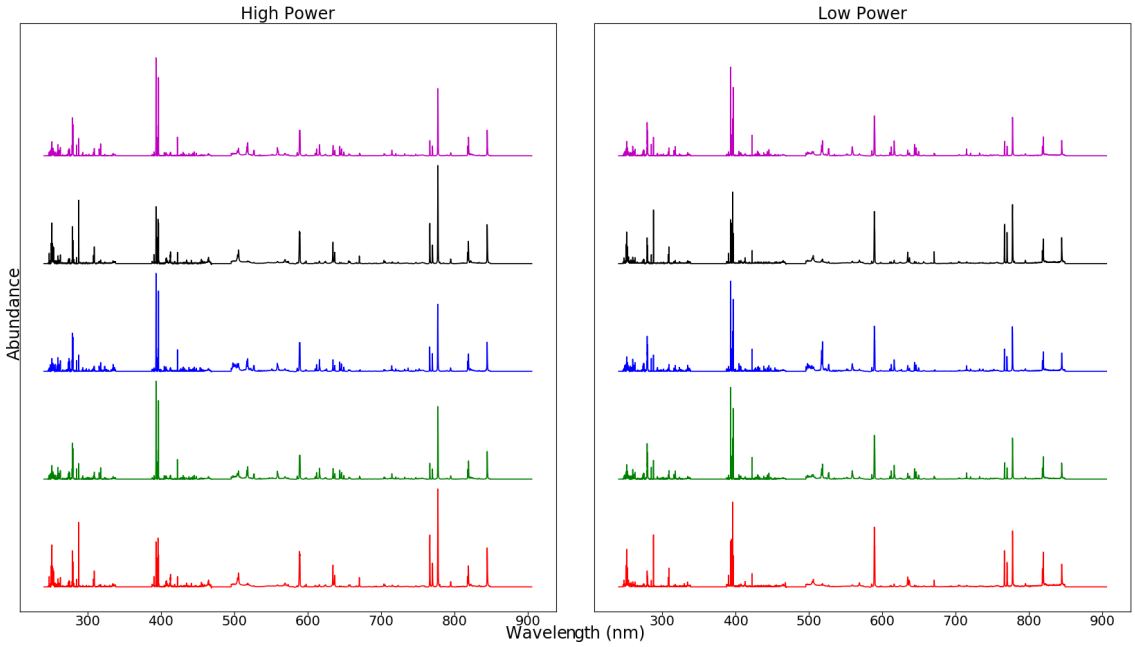


Figure 3.6: Five mineral spectra selected at random from the LIBS data sets. The left hand side shows the spectra recorded with a high power laser, and the right hand side shows the corresponding spectra recorded at a low power.

The task of this experiment was to align the set of low power spectra with the set of high power spectra. A low power spectrum was considered correctly aligned if the corresponding high power spectrum was within its 5-nearest neighbors in the embedded space.

For all models evaluated, the correspondence weight was set to $\mu = 0.8$, based upon the ratio of train/test data. All competing models required an additional nearest neighbor hyperparameter. This hyperparameter was optimized using grid search and cross validation. For affine matching and Procrustes alignment the number of neighbors used was $k = 10$, and for traditional manifold alignment $k = 4$. For all of these competing methods, a binary weight was used in the graph construction because it proved more accurate than the heat kernel for this experiment.

The 5-fold cross validation results are shown in Figure 3.7. In each iteration, the training samples were provided as correspondences and the test samples were used

for evaluation. LRA outperformed all other models tested, achieving an accuracy of 67.8% at $d = 16$. The next best performing model, KLRA with a degree 3 polynomial kernel, had an accuracy of 56.3% at $d = 66$. The three non-LRA methods all performed comparably well, manifold alignment 42.0%, Procrustes alignment 38.7%, and affine matching 36.5%.

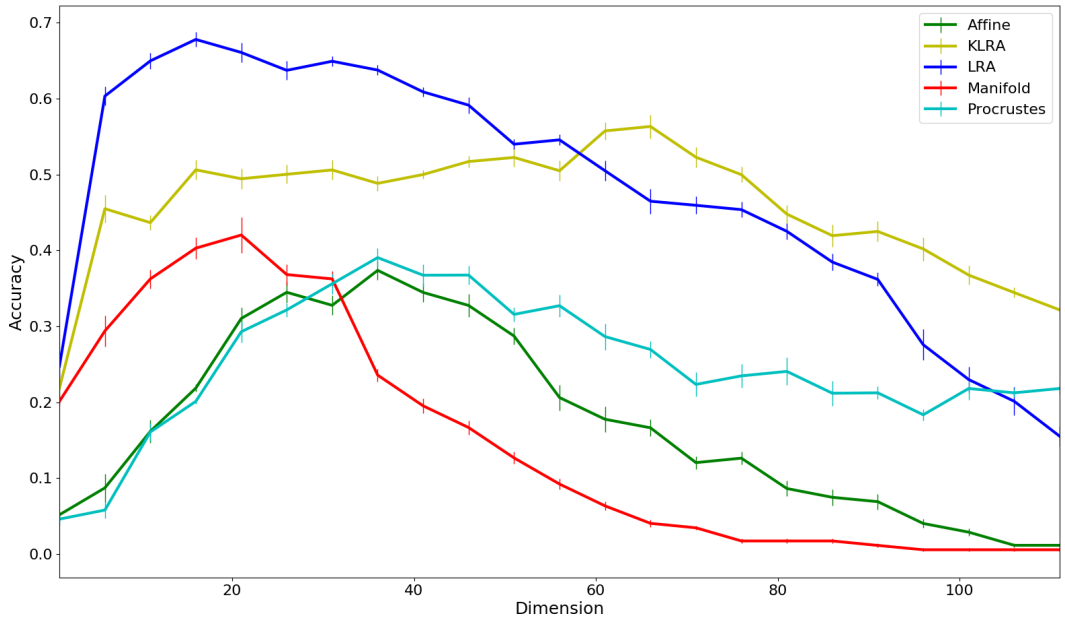


Figure 3.7: Cross validation results of the hi/low power LIBS spectra alignment experiment, including bars indicating the standard error of the mean.

In this last test, it was assumed that all of the spectra were recorded at both power settings, but in reality CT is often used when only a portion of the sample set is recorded under both conditions. For example, a researcher may have a large database recorded at high power that he or she uses to fit a regression model for predicting the chemical compositions (% weight) of the spectra. As commonly occurs, the researcher also has a smaller *calibration set* recorded at both high and low powers. Unfortunately, an unforeseen instrument malfunction occurs allowing the spectrometer to only use low power. To predict subsequent low power spectra using the high power database, an alignment must be calculated.

To simulate this situation, an alignment was calculated using 30 samples at both powers (the calibration set), 50 samples at only high power (the large database), and 20 samples at only low power. The 20 low power samples represent the *out-of-sample* spectra recorded after instrument malfunction. To note, this results in a non-square correspondence matrix $C^{(X,Y)}$.

Next, a multivariate linear regression model was trained to predict 10 major elements of the minerals (e.g., SiO_2, Al_2O_3, CaO) using the embedded high power database and the embedded calibration samples. To evaluate the regression model, the compositions of the 20 embedded low power spectra were predicted and compared to ground-truth composition values.

Setting $d = 8$, the experiment was repeated 30 times with randomized sets. The regression model trained on LRA achieved on average a 1.8%, 4.8%, and 8.1% improvement in RMSEP over affine matching, Procrustes alignment, and traditional manifold alignment, respectively. This shows that the high accuracy of LRA in alignment translates to improved performance in the final predictive model.

3.6.2 European Parliament Proceedings

In this second set of experiments, we used the transcribed proceedings of the European Parliament [48] for a standard cross-language document retrieval task. The task is simply stated: given a document in one language, find its matching document in the second language. The parliament corpus was collected between April 1996 and November 2011 and transcribed into 21 European languages. In the corpus, each utterance of a speaker was transcribed into paragraphs of typically 2-5 sentences. This data set is commonly used when comparing manifold alignment algorithms [81, 84].

In the first experiment, we align the German corpus with the English corpus, and in the second experiment we align the Italian corpus with the English corpus. We chose these languages because each had approximately 1.9 million sentence pairs.

To represent the utterances, a bag-of-words model was used, where the 2500 most frequently occurring words were considered, after filtering for stop-words. Unlike methods like [31] and [63] that used domain knowledge in their model preprocessing, we used a simple statistical model to compare the different alignment methods more directly. To pare down the data set for efficient experimentation, only sentences with more than 45 words were used, resulting in a subset of 1200 sentence pairs for both English-German and English-Italian experiments. For accurate method comparison, we used 5-fold cross validation. In each fold, 80% of the sentence correspondences were provided and the remaining 20% of the sentences were used for evaluation. To evaluate a sentence alignment, we define a correct translation as a sentence embedding where the true correspondence pair appears within the 10-nearest neighbors in the embedded space.

All LRA methods used the same default correspondence weight $\mu = 0.8$. KLRA used an RBF kernel with the bandwidth set to $\gamma = 0.1$. Grid search and cross validation were used to tune the number of nearest neighbors for all competing models. For affine matching and Procrustes alignment $k = 125$, and for manifold alignment $k = 5$.

Results of the text alignment test are shown in Figure 3.8. KLRA outperformed all other evaluated models in the English-German experiment and the English-Italian experiment, with an accuracy of 94.3% at embedding dimension $d = 80$ and 96.8%, respectively. Traditional LRA performed a few percent worse than KLRA on both the English-German and the English-Italian experiments, with an accuracy of 90.1% and 93.3%, respectively. In contrast, the three non-LRA methods, affine matching having the highest accuracy, performed about half as well as LRA.

Traditional manifold alignment was clearly the worst-performing model. Affine matching and Procrustes alignment are both two-step algorithms in the sense that they rely on a second transformation after the embedding step. In contrast, manifold

alignment and LRA are one-step algorithms that incorporate these constraints into their embeddings and so could be seen to place more importance on their ability to recover the shared manifold of the data sets. This skewed performance between methods suggests that the corpora are drawn from mixtures of manifolds.

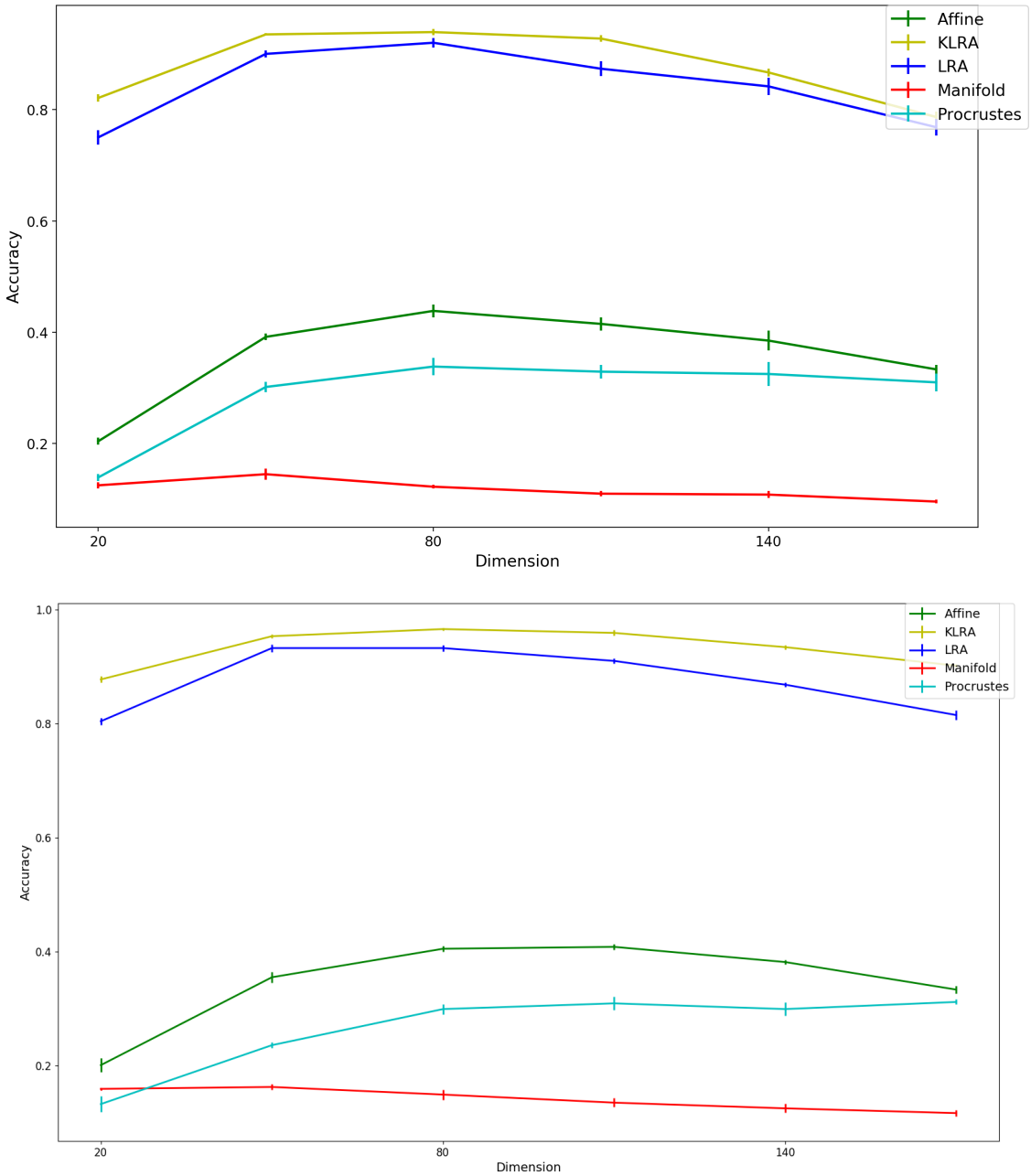


Figure 3.8: Cross validation results of the EU parallel corpus experiments for the English-German sentence pairs (top) and the English-Italian sentence pairs (bottom), including bars indicating the standard error of the mean.

3.6.3 Clustering News Topics

The effectiveness of clustering with LRA on real-world problems is demonstrated by a cross-lingual clustering experiment. The data set used was a publicly available collection of news documents from Reuters where each document is available in five languages: English, French, German, Italian, and Spanish [2]. Each article has been categorized, and for this experiment, six of the most populous categories were selected: C15, CCAT, E21, ECAT, GCAT, and M11. The whole data set consists of 12-30K documents per language (depending on the language) and 13-20k documents per class. To make the problem more computationally tractable, 300 documents for each language comparison were randomly selected from each category totaling 1800 samples. The data were represented as a bag of words model and then transformed using a TF-IDF-based weighting scheme. The top 11-35K words were used for each language.

To measure the quality of the clustering, two metrics were used: the adjusted rand score (ARS) and the adjusted mutual information score (AMIS). The traditional rand score (or index) is like a measure of accuracy between two clusterings. Standard accuracy cannot be used because the true clustering is only known up to a permutation of the labels. The ARS is a corrected-for-chance version of the rand score. ARS values range from $(-\infty, 1]$, where 1 is a perfect score. Mutual information (MI) is an information theoretic measure of the mutual dependency between two random variables, where the clusterings are the variables. The AMIS is a corrected-for-chance version of the MI score. The AMIS is 1 when the clusterings are equal and 0 when the MI between the clusterings equals to that expected by chance.

A total of four experiments were run, where each non-English language was aligned and clustered with English. Each experiment was repeated with 10 random trials, and one standard deviation is reported for both metrics. Two versions of LRA were included, one where 25% of the correspondences were provided (LRA-25%) and one

where 50% of the correspondences were provided (LRA-50%). These were compared to two alternate methods, latent semantic analysis (LSA) [40, 43] and spectral clustering (SC) [73]. Because the data were provided in only TF-IDF format, standard methods like latent Dirichlet allocation (LDA) [7] could not be used. Also, because SC required all of the data to be in the same representation, LSA was used to first transform the samples in both languages to a 100-dimensional space, and then spectral clustering was applied. For all algorithms evaluated, the clustering was performed in 10-dimensional space. Results of the experiments are listed in Table 3.1. Overall, SC was the worst performing method. This is potentially due to the required LSA preprocessing. Also, similar to LSA, SC does not first align the data sets and cannot use the correspondences in its embedding. So when combining different data sets, neighborhood-based methods like SC can have a difficult time accurately measuring distances between samples across data sets. LSA generally performs better than SC, but it was a method made specifically for classifying and clustering text data. However, LSA still does not match the performance of either LRA model. LRA-50% has the highest scores for both metrics on all four experiments. As expected, it consistently outperformed LRA-25%; however, there were diminishing returns on the number of correspondences, whereby LRA-50% did not perform twice as well as LRA-25%.

3.6.4 Actively Learning Synthetic Correspondences

To examine the effectiveness of the τ-score, it was first applied to two synthetic examples. First, a simple two-data set synthetic example was examined, where the ambient dimension was 2 and the intrinsic dimension was 1. The first data set $X^{(1)}$ was composed of samples drawn uniformly from two crossed 1-dimensional parameterized sine curves $y = f(x) = (0.2)\sin(kx)$ with varying periods $k = 2, 20$. The second data set $X^{(2)}$ was a copy of the first that has been rotated, translate, and scaled.

Adjusted Mutual Information Score				
	French	German	Italian	Spanish
LRA-25%	0.202 ± 0.036	0.194 ± 0.014	0.150 ± 0.052	0.190 ± 0.053
LRA-50%	0.247 ± 0.038	0.244 ± 0.034	0.216 ± 0.045	0.225 ± 0.032
LSA	0.137 ± 0.017	0.147 ± 0.010	0.140 ± 0.010	0.117 ± 0.017
SC	0.076 ± 0.007	0.073 ± 0.001	0.083 ± 0.004	0.018 ± 0.001
Adjusted Rand Index Score				
	French	German	Italian	Spanish
LRA-25%	0.133 ± 0.044	0.135 ± 0.039	0.105 ± 0.047	0.127 ± 0.049
LRA-50%	0.174 ± 0.036	0.177 ± 0.026	0.130 ± 0.045	0.153 ± 0.026
LSA	0.068 ± 0.009	0.083 ± 0.013	0.079 ± 0.007	0.048 ± 0.010
SC	0.015 ± 0.003	0.011 ± 0.000	0.016 ± 0.000	0.080 ± 0.003

Table 3.1: Results from clustering Reuters cross-lingual data set comparing low rank alignment given 25% correspondences (LRA-25%) and given 50% correspondences (LRA-50%), latent semantic analysis (LSA), and spectral clustering (SC). For both metrics, higher is better.

In this experiment, the accuracy of the alignment was measured as the number of correspondences c was varied. As a baseline comparison, the τ-score was compared to randomly selected correspondences. For each setting of c, the random baseline experiment was repeated 1000 times. It was also compared against the Kennard-Stone (KS) representative subset selection algorithm [47]. In the KS algorithm, the sample closest (in Euclidean space) to the mean was first selected for the training set, then the set was constructed iteratively, where the next sample chosen was the one farthest from the closest current training set sample [18]. To measure the accuracy of the alignment, because both data sets used the same parameterization, meaning all samples are technically in correspondence but not revealed to the algorithm, the accuracy of the alignment can be directly measured as

$$error(F) = \frac{1}{k} \sum_{i \neq j}^{k} \left\| F^{(i)} - F^{(j)} \right\|, \tag{3.38}$$

where $F^{(1)}$ and $F^{(2)}$ are the LRA embeddings of $X^{(1)}$ and $X^{(2)}$, respectively.

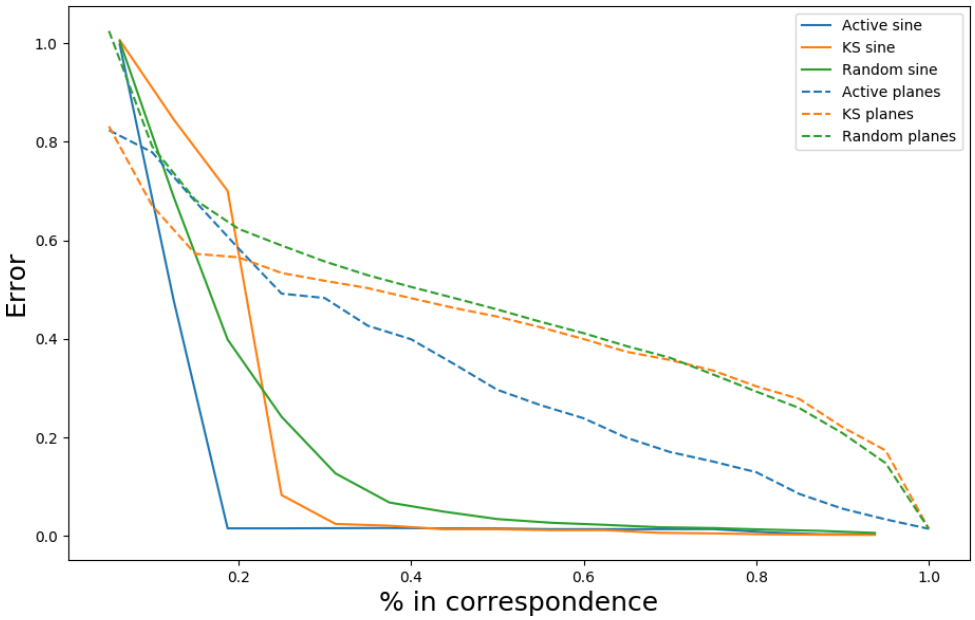


Figure 3.9: Results of the two active learning experiments performed on synthetic data. Three methods were used to actively learn correspondences for low rank alignment (LRA): our active learning algorithm (Active), the Kennard-Stone representative subset selection algorithm (KS), and random selection. The error is defined in equation 3.38.

Results of this experiment are displayed in Figure 3.9. In this simple sine experiment, active learning outperformed the competing methods, for all numbers of correspondences c. The drop in error was so dramatic because the underlying manifolds were simple 1-D line segments. Active learning requested the four ends of the segments first, and it was able to fully disentangle the manifolds and embed them with little loss after only three correspondences. In comparison, because the KS algorithm iteratively selects samples dissimilar to those already selected, it did not choose the segment ends in the second manifold. The random selection method decayed more smoothly and slowly than the other methods, indicating that the underlying LRA operation performs smoothly when randomly given correspondences.

The second synthetic experiment was aligning two sets of intersecting planes in 3D. In each set, a plane was uniformly randomly sampled from the set of triplets:

$$\{(x, y, 0) : x, y \in [0, 1)\}.$$

Afterward, a second plane with the same parameterization was then created and transformed by random rotations and translations. This problem was difficult because all four of the planes were overlapping and intersecting one another. This experiment was repeated for 100 random trials. Results of the experiments are shown in Figure 3.38. Because of the much larger sample size than the previous example, the error curves for all of the methods evaluated were smoother. Initially when $c < 25\%$, because the KS method uniformly selects points in a grid-like fashion from the planes, it performed better at selecting a representative set. With a smoother manifold than the previous example, there were fewer intuitive key samples, so the active learning approach did not surpass the competing methods until $c >= 25\%$. The active learning method tended to select central points on the planes and diffuse out towards the edges. Both KS and random had nearly identical performance, and both lagging considerably behind the active method in performance.

3.6.5 Learning Cross-lingual Correspondences

To examine the effectiveness of this active learning method on real-world data, the technique was applied to the same multi-lingual European Parliament Proceedings data set used in section 3.6.2. Two alignment experiments were performed: first a set of English documents was aligned with their matching German documents, and second, the experiment was repeated aligning a set of English and Italian documents. In each experiment, 1000 documents per language were randomly selected from the corpus. The same normalized bag-of-word representation (as in section 3.6.2) was used for both experiments. The number of correspondences was varied from 10% to

100%, and the same direct *error* from equation 3.38 was used to measure the quality of the alignment. The KS selection algorithm computation was much too slow to be included in this experiment, so instead the active learning method was only compared to the random correspondence baseline. For each setting of correspondence number (e.g., $c = 10\%$), the average of 10 random trials was calculated, where a new set of documents was selected in each trial.

Results from this experiment are listed in Table 3.10. The τ-score active learning scheme proved effective at this real-world domain. In both German-English and Italian-English experiments, the active learning model significantly outperformed correspondences selected at random. The English-Italian alignment proved more amenable to active learning, where for $c >= 50\%$ the error of the baseline double or more the error of the actively selected correspondences. The English-German alignment showed a similar pattern of performance, with the exception of $c = 40\%$, where a dip in performance from the active learning method matched that of the baseline method.

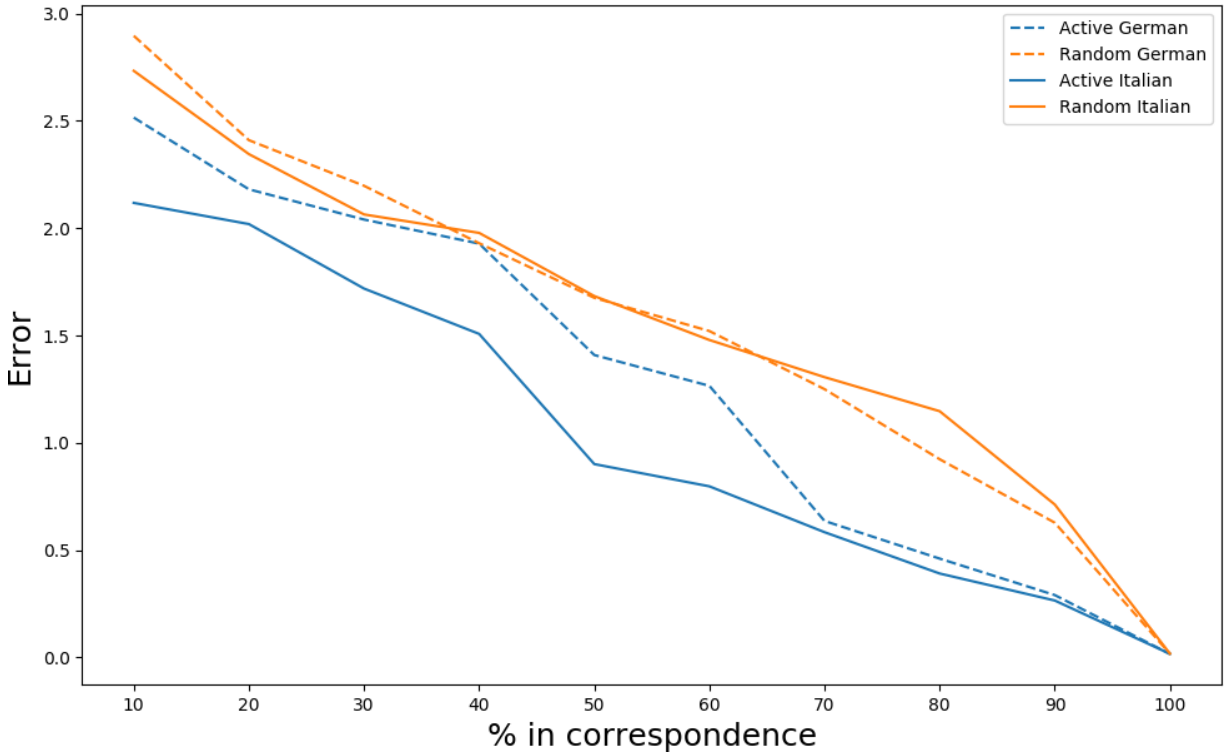


Figure 3.10: Results of the active learning cross-language experiment, using European Parliament proceedings. Low rank alignment is to align English and German documents and English and Italian documents. The error is defined in equation 3.38.

3.7 Remarks

This chapter presented a novel framework for manifold alignment that can align data sets drawn from a mixture of manifolds. Unlike previous manifold alignment algorithms that rely on nearest neighbor graph construction, LRA instead uses a low rank matrix constraint to calculate its reconstruction weight matrix, which was demonstrated to be less prone to short-circuit connections. Algorithms for both linear and non-linear manifolds were presented. A small modification was introduced to improve its performance for downstream clustering. Lastly, a method for actively selecting the most beneficially correspondences was described.

CHAPTER 4

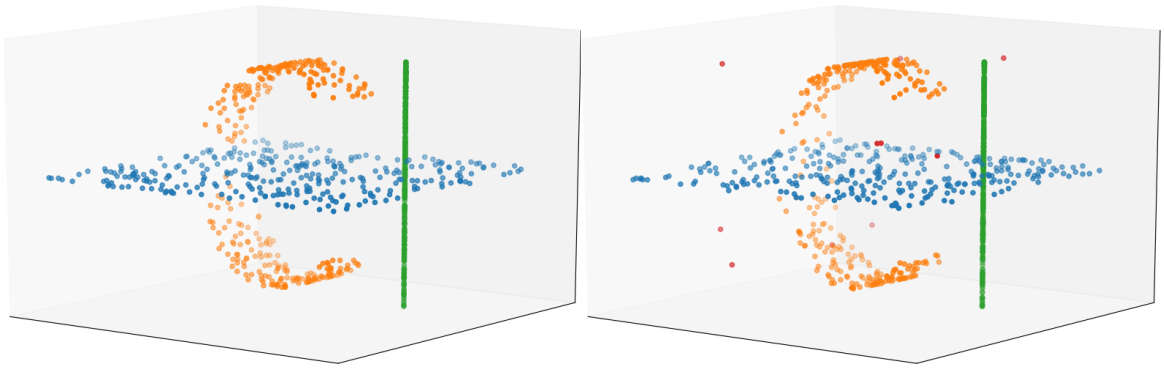
ALIGNING MANIFOLDS UNDER NOISY CONDITIONS

Real-world data are often corrupted or noisy in nature, and these nuisances can make aligning data sets difficult. While there are many sources of noise, a few types of noise are described in this chapter, along with specific algorithms for solving them. These robust modifications require a new iterative LRA algorithm.

Two general forms of noise are illustrated in Figure 4.1. First, there is general set-wide noise, which can range from mild to severe, where the data are slightly perturbed around the manifolds. Second, there is sample-specific noise that are outliers from the data distribution. Local neighborhood-based manifold methods are especially susceptible to noisy samples and outliers, and a number of methods have been proposed to solve this problem [15, 36].

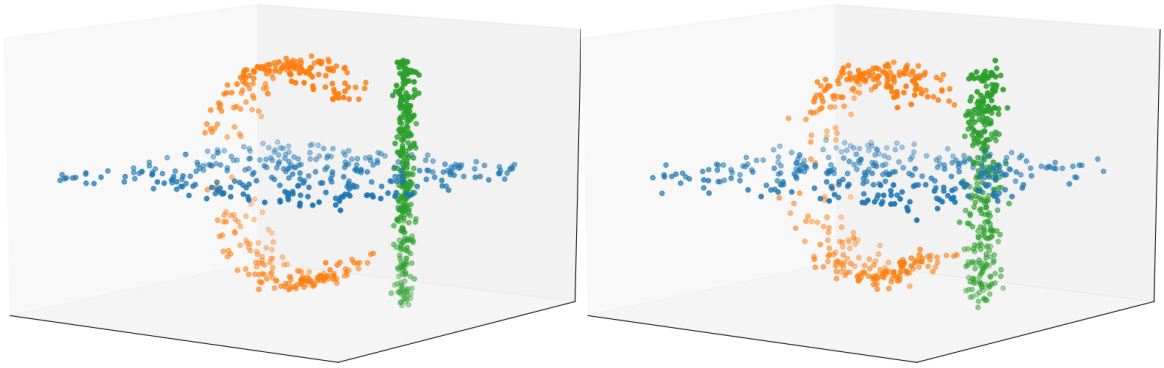
4.1 Sparse and Low Rank Alignment

The reconstruction weight matrix R is composed of dense blocks $R^{(X_1)}, \ldots, R^{(X_m)}$. These dense weight matrices can form a fully connected graph, where all samples across all manifolds, not just neighbors, are connected. Ideally, only neighbors on the same manifold are connected. This implies that given the proper row ordering, whereby samples from the same manifold are stacked adjacent in blocks of the matrix, the resulting data set reconstruction matrix $R^{(X_k)}$ should be block diagonal. The off-diagonal zero entries show there are no short-circuits between the multiple manifolds. To make the math of this section more readable, we will use $\mathcal{R}$ in place of $R^{(X_k)}$ because this section applies on a per-data set basis. This should not be confused with



(a) A clean sampling from a mixture of three intersecting manifolds.

(b) Examples of outliers are points colored in red.



(c) A moderate amount of general noise.

(d) A severe amount of general noise.

Figure 4.1: Examples of the types of noise encountered during manifold.

the block-diagonally constructed joint data reconstruction matrix R from equation 3.6. Using the standard LRA algorithm, the reconstruction matrix found in step one (equation 3.1) may contain small amounts of noise in the off diagonal entries. To eliminate these noisy short-circuits from the LRA algorithm, an additional $\|\cdot\|_1$ penalty can be added to the reconstruction matrix,

$$\min_{\mathcal{R}} \frac{1}{2} \|X - \mathcal{R}X\|_F^2 + \lambda \|\mathcal{R}\|_* + \mu \|\mathcal{R}\|_1 , \tag{4.1}$$

where μ is a non-negative hyperparameter controlling the sparsity of the reconstruction matrix $\mathcal{R}$.

Equation 4.1 is convex but non-differentiable, and while the low-rank penalty alone has a closed form solution, unfortunately the sparse low-rank penalty has no known closed form solution. Instead, the equation can be optimized using the ADMM algorithm [11]. Forward-backward splitting and proximal descent methods could also be used [68], but in practice, ADMM converged much faster than these two methods.

The generic ADMM optimization problem considers convex functions f and g and minimizes the constrained equation

$$\min f(x) + g(z) \text{ subject to } Ax + Bz = c. \tag{4.2}$$

In the case of sparse LRA, it is $f(X) = \frac{1}{2}\|X - \mathcal{R}X\|_F^2$, $g(Z) = \lambda\|Z\|_* + \mu\|Z\|_1$, with the constraint $\mathcal{R} = Z$. Constructing the loss function by splitting variables $\mathcal{R}$ and Z allows the differentiable and non-differentiable terms of equation 4.1 to be optimized separately. To impose the equality constraint $\mathcal{R} = Z$ the *augmented Lagrangian* L_ρ is formed

$$L_\rho(\mathcal{R}, Z, \mathcal{L}) = \frac{1}{2}\|X - \mathcal{R}X\|^2 + \lambda\|Z\|_* + \mu\|Z\|_1 + \langle \mathcal{L}, \mathcal{R} - Z\rangle + \frac{\rho}{2}\|\mathcal{R} - Z\|^2, \tag{4.3}$$

where $\mathcal{L}$ is the *Lagrange dual* variable to enforce the equality constraint and $\rho > 0$ is a penalty parameter controlling the rate of convergence by enforcing equality.

To optimize equation 4.3, the ADMM algorithm iterates over three steps:

$$\mathcal{R}^{k+1} = \arg\min_{\mathcal{R}} L_\rho(\mathcal{R}, Z^k, \mathcal{L}^k) \tag{4.4}$$

$$Z^{k+1} = \arg\min_{Z} L_\rho(\mathcal{R}^{k+1}, Z, \mathcal{L}^k) \tag{4.5}$$

$$\mathcal{L}^{k+1} = \mathcal{L}^k + \rho(\mathcal{R}^{k+1} - Z^{k+1}). \tag{4.6}$$

The first step (4.4) is a minimization of $\mathcal{R}$. This can be solved in closed form using standard matrix calculus techniques, resulting in

$$\mathcal{R}^{k+1} = \left(X^\top X + \rho I\right)^{-1} \left(X^\top X + \rho Z^k - \mathcal{L}^k\right). \tag{4.7}$$

The second step (4.5) is a minimization of Z. This is non-differentiable, so standard gradient methods cannot be used. Instead, the *proximal operator* $\text{prox}_{\lambda g}(x)$ of the function $g : \mathcal{D} \mapsto \mathbb{R}$ is used,

$$\text{prox}_{\lambda g}(x) = \underset{u \in \mathcal{D}}{\text{argmin}} \left(g(u) + \frac{1}{2\lambda} \|u - x\|_2^2 \right), \tag{4.8}$$

where $\lambda > 0$ is a mixing parameter controlling how far x is allow to stray from u to minimize g. The proximal operator (or mapping) can be interpreted as a generalization of the projection operator. If x is outside the domain $\mathcal{D}$ of g, then $\text{prox}_{\lambda g}(x)$ will map x to a point in $\mathcal{D}$ that also minimizes g. Moreover, if g is the indicator function of a set $\mathcal{C}$, then $\text{prox}_{\lambda g}(x)$ is the Euclidean projection onto $\mathcal{C}$.

The proximal operator can also be interpreted as a type of gradient descent operator because it minimizes the function. Using this, the minimization problem in equation 4.5 is rewritten as

$$\begin{aligned} Z^{k+1} &= \arg\min_Z \left(\lambda \|Z\|_* + \mu \|Z\|_1 - trace\left(\mathcal{L}^{k\top} Z\right) + \frac{\rho}{2} \left\|\mathcal{R}^{k+1} - Z\right\|_F^2 \right) \\ &= \arg\min_Z \left(g(Z) + \frac{\rho}{2} \left\| \mathcal{R}^{k+1} + \frac{1}{\rho}\mathcal{L}^k - Z \right\|_F^2 \right) \\ &= \text{prox}_{\frac{g}{\rho}} \left(\mathcal{R}^{k+1} + \frac{1}{\rho}\mathcal{L}^k \right), \end{aligned} \tag{4.9}$$

where $trace(\cdot)$ is the matrix trace. Therefore, to optimize 4.5, only the proximal operator of g need be calculated.

The proximal operator of a vector norm $h = \|\cdot\|$ is $\text{prox}_{\lambda h}(x) = x - \lambda \prod_{\mathcal{B}}(x/\lambda)$, where $\prod_{\mathcal{B}}$ is the projection onto the unit ball $\mathcal{B}$ of the norm. For a proof see section

6.5 of [60]. From this, the proximal operator for the l_1 norm, called *soft thresholding*, can be deduced. The soft thresholding operator is defined as

$$\text{prox}_{\mu\|\cdot\|_1}(Z) = \begin{cases} Z_{i,j} - \mu, & Z_{i,j} > \mu \\ 0, & |Zi,j| \leq \mu \\ Z_{i,j} + \mu, & Z_{i,j} < -\mu \end{cases}. \tag{4.10}$$

Here the vector norm is applied entry-wise to the matrix Z. This piece-wise definition can be defined as a single function

$$\text{prox}_{\mu\|\cdot\|_1}(Z) = \text{sign}(Z) \cdot \max(0, |Z| - \mu). \tag{4.11}$$

The nuclear norm is a Schatten matrix norm, i.e., $\|Z\|_* = \|diag(\Sigma)\|_1$ for $\text{SVD}(Z) = U\Sigma V^\top$, and so is equal to the proximal operator of the l_1-norm applied to the vector of singular values in the SVD,

$$\text{prox}_{\lambda\|\cdot\|_*}(Z) = U\left(\text{prox}_{\lambda\|\cdot\|_1}(\Sigma)\right)V^\top. \tag{4.12}$$

Because $\|\cdot\|_*$ and $\|\cdot\|_1$ are both norms, they are necessarily closed convex, positive homogeneous functions. Therefore, Theorem 4 of Yu [96] applies, and the sum of operators is equal to their composition,

$$\text{prox}_{\mu\|\cdot\|_1+\lambda\|\cdot\|_*} = \text{prox}_{\mu\|\cdot\|_1} \circ \text{prox}_{\lambda\|\cdot\|_*}. \tag{4.13}$$

Thus, step (4.5) reduces to

$$Z^{k+1} = \text{prox}_{\mu\|\cdot\|_1}\left(\text{prox}_{\lambda\|\cdot\|_*}\left(\mathcal{R}^{k+1} + \frac{1}{\rho}\mathcal{L}^k\right)\right). \tag{4.14}$$

To preserve the sparsity of Z, it is recommended that the composition be performed in this order. Furthermore, while Z converges to $\mathcal{R}$, computationally in practice it is often better to return Z to preserve zero-value entries.

The last step (4.6) is an update of the Lagrange dual variable $\mathcal{L}$. These three steps are repeated until the variables Z and R converge. A simple test for convergence is if

$$\left\|Z^{k+1} - Z^k\right\| / \left\|Z^{k+1}\right\| < \epsilon_{tol} \quad \text{and} \quad \left\|R^{k+1} - R^k\right\| / \left\|R^{k+1}\right\| < \epsilon_{tol} \tag{4.15}$$

for some small tolerance like $\epsilon_{tol} = 10^{-4}$. Alternately, convergence of the primal residual $\left\|R^{k+1} - Z^{k+1}\right\|$ and the dual residual $\left\|\rho(Z^{k+1} - Z^k)\right\|$ can be used for stopping criteria. See section 3.3.1 of [11] for greater detail.

The sparse LRA algorithm differs from the standard LRA algorithm only in the reconstruction step. Algorithm 4 details this step. The runtime of this algorithm is dominated by the singular value decomposition in the SVT step, which is cubic time. Note that the most outer loop can be easily parallelized between CPUs or computers.

In practice, ADMM typically converges quickly to a good solution. To decrease the computational burden of solving equation 4.4 in each iteration, the (symmetric, positive semi-definite) term $X^\top X + \rho I$ may be decomposed into triangular matrices using the Cholesky decomposition. Performing this operation once before gradient descent makes all subsequent calculations of R^{k+1} more efficient. Instead of iterating through the subsets $X^{(i)}$ sequentially, the gradient descent calculations of $R^{(i)}$ can be naturally parallelized into k current processes. For extremely high-dimensional data with thousands of features or more, it may be computationally advantageous to distribute the problem across features. This is detailed in section 8.3 of [11].

4.2 Robust LRA

The last section introduced a version of LRA that reduces noisy short-circuit connections in the reconstruction matrix, which works well at reducing noise in manifold

Algorithm 4: Sparse LRA Reconstruction Step

Input : $X = \text{diag}\left(\{X^{(i)} : i \in (1, \ldots, k)\}\right)$, the block diagonal data matrix,
μ, weight of the sparsity norm,
λ, weight of the low-rank norm, default 1.
Output: R, a k block diagonal reconstruction matrix.
begin
  for $X^{(i)} \in X$ **do**
    Initialize $R^{(i)}, Z, \mathcal{L} \leftarrow \mathbf{0}, \rho \leftarrow 1\text{e}^{-5}$.
    while *not converged* **do**
      $R^{(i)} \leftarrow \left(X^{(i)^\top} X^{(i)} + \rho I\right)^{-1} \left(X^{(i)^\top} X^{(i)} + \rho Z^k - \mathcal{L}^k\right)$
      $U, \Sigma, V^\top \leftarrow \text{SVT}\left(R^{(i)} + \mathcal{L}/\rho\right)$
      $Z \leftarrow U \ \text{diag}\left(\max\left(0, \Sigma - \lambda\right)\right) \ V^\top$
      $Z \leftarrow \text{sign}(Z) \cdot \max\left(0, |Z| - \mu\right)$
      $\mathcal{L} \leftarrow \mathcal{L} + \rho\left(R^{(i)} - Z\right)$
    end
  end
  $R \leftarrow \text{diag}\left(\{R^{(i)} : i \in (1, \ldots, k)\}\right)$
end

intersection areas. However, this does not directly address the case where the data are corrupted by outliers or sampled noisily from their underlying manifolds. In this section, a robust version of LRA is presented that corrects this problem by directly modeling the error term. Modeling the error removes it from the reconstruction problem and may also provides insight to the researcher about the nature of their data collection techniques.

To dissect the noise from the data, in step 1 of LRA, for each data set X_i for $i = 1, \ldots, k$, equation 3.1 is replaced with

$$\arg\min_{R_i} \|X_i - (R_i X_i + E_i)\|_F^2 + \alpha \|R_i\|_* + \beta \left\|E_i^\top\right\|_{2,1}, \tag{4.16}$$

where $E_i \in \mathbb{R}^{N_i \times D_i}$ is an error term modeling the noise. For an arbitrary matrix $A \in \mathbb{R}^{r \times p}$, its $l_{2,1}$-norm is the defined as

$$\|A\|_{2,1} = \sum_{i=1}^{r} \sqrt{\sum_{j=1}^{p} A_{ij}^2}. \tag{4.17}$$

This is simply the sum of the Euclidean norm of the columns of A, and the penalty enforces column-wise smoothness and sparsity.

To better understand the effects of the $l_{2,1}$ norm penalty, a synthetic example is shown in Figure 4.2. Let $A \in \mathbb{R}^{20\times 20}$ be a matrix whose first sixteen columns $[a_1 a_2 \cdots a_{16}]$ are drawn i.i.d. and last four columns $[a_{17} \cdots a_{20}]$ are outliers. To optimize equation 4.16 and enforce the penalty, the proximal operator $\text{prox}_{l_{2,1}}(A)$ will be used, where

$$\text{prox}_{l_{2,1}}(A) = \min_{Z} \alpha ||Z||_{2,1} + \frac{1}{2}||Z - A||_F^2. \tag{4.18}$$

The optimal solution Z is defined column-wise such that the i-th column is

$$Z(:,i) = \begin{cases} \frac{\|a_i\|-\alpha}{\|a_i\|} a_i, & \text{if } \alpha < \|a_i\| \\ 0, & \text{otherwise.} \end{cases} \tag{4.19}$$

After applying the proximal operator to A, its last four outlier columns were driven to zero, and the values between rows were smoothed and made less saturated by high value peaks.

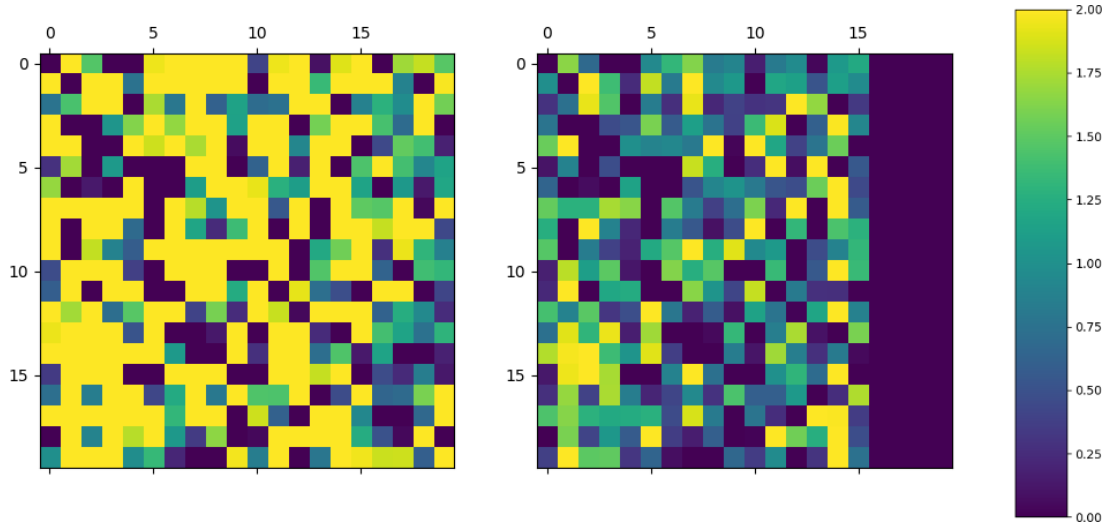


Figure 4.2: On the left is a matrix A where the first 16 columns are drawn i.i.d and the last 4 columns are outliers. On the right is the matrix $\text{prox}_{l_{2,1}}(A)$ after applying the $l_{2,1}$ proximal operator. Notice that the outliers have been eliminated and between the rows have been smoothed.

An extended version of ADMM can be used to optimize 4.16 that simultaneously solves for both the reconstruction term R and the error term E. The extension requires two different Lagrange multipliers, a new multiplier $\mathcal{L}_1$ that enforces the constraint $E = X - RX$, and the usual multiplier $\mathcal{L}_1$ that enforces the constraint $R = Z$. As was done in equation 4.3, the augmented Lagrangian is formulated as

$$\begin{aligned} L(Z, R, E, \mathcal{L}_1, \mathcal{L}_2) = \lambda \|Z\|_* + \mu \left\|E^\top\right\|_{2,1} \\ + \langle \mathcal{L}_1, X - RX - E \rangle + \frac{\rho}{2} \|X - RX - E\|_F^2 \\ + \langle \mathcal{L}_2, R - Z \rangle + \frac{\rho}{2} \|R - Z\|_F^2 . \end{aligned} \tag{4.20}$$

To minimize L using ADMM, each of the variables being solved $R, Z, E, \mathcal{L}_1, \mathcal{L}_2$ must must be updated in each iteration. The Z-update step is derived in a similar way as equation 4.9; however, in the second line, g is the operator $g(Z) = \|Z\|_*$ defined in equation 4.12. Define singular-value thresholding as

$$\mathrm{SVT}_{\mu}(X) = U \max\left(0, \Sigma - \mu\right) V^{\top} \text{ for } U, \Sigma, V^{\top} = \mathrm{SVD}\left(X\right). \tag{4.21}$$

Then the Z^{k+1} update step is defined as

$$\begin{aligned}
Z^{k+1} &= \arg\min_{Z} \lambda \left\|Z\right\|_{*} + \left\langle \mathcal{L}_2, R - Z \right\rangle + \frac{\rho}{2} \left\|R - Z\right\|_F^2 \\
&= \arg\min_{Z} \lambda \left\|Z\right\|_{*} + \frac{\rho}{2} \mathrm{tr}\left(ZZ^{\top} - 2RZ^{\top} - \frac{2}{\rho}\mathcal{L}_2 Z^{\top}\right) \\
&= \arg\min_{Z} \lambda \left\|Z\right\|_{*} + \frac{\rho}{2} \left\|Z - R - \frac{1}{\rho}\mathcal{L}_2\right\|_F^2 \\
&= \mathrm{prox}_{\frac{1}{\rho}\|\cdot\|_{*}}\left(R + \frac{1}{\rho}\mathcal{L}_2\right) \\
&= \mathrm{SVT}_{\rho^{-1}}\left(R + \frac{1}{\rho}\mathcal{L}_2\right).
\end{aligned} \tag{4.22}$$

The update step E^{k+1} is calculated, and it too will result in a proximal operator update. Collecting the applicable terms from equation 4.20 results in

$$\begin{aligned}
E^{k+1} &= \arg\min_{E} \mu \left\|E^{\top}\right\|_{2,1} + \left\langle \mathcal{L}_1, X - RX - E \right\rangle + \frac{\rho}{2} \left\|X - RX - E\right\|_F^2 \\
&= \arg\min_{E} \mu \left\|E^{\top}\right\|_{2,1} + \frac{\rho}{2} \mathrm{tr}\left(E^{\top}E + 2X^{\top}R^{\top}E - 2X^{\top}E - \frac{2}{\rho}\mathcal{L}_1^{\top}E\right) \\
&= \arg\min_{E} \frac{\mu}{\rho} \left\|E^{\top}\right\|_{2,1} + \frac{1}{2} \left\|E^{\top} - X^{\top} + X^{\top}R^{\top} - \frac{1}{\rho}\mathcal{L}_1^{\top}\right\|_F^2 \\
&= \mathrm{prox}_{\frac{\mu}{\rho}\|\cdot\|_{2,1}}\left(X^{\top} - X^{\top}R^{\top} + \frac{1}{\rho}\mathcal{L}_1^{\top}\right).
\end{aligned} \tag{4.23}$$

In equation 4.19, the $\left\|\cdot\right\|_{2,1}$ proximal operator is defined.

The update step R^{k+1} is derived using traditional calculus methods. First, the terms containing R are collected from equation 4.20 and expanded,

$$\begin{aligned}
R^{k+1} &= \arg\min_{R} \langle \mathcal{L}_1, X - RX - E \rangle + \frac{1}{2} \left\|X - RX - E\right\|_F^2 + \left\langle \mathcal{L}_2, R - Z \right\rangle + \frac{\rho}{2} \left\|R - Z\right\|_F^2 \\
&= \arg\min_{R} \mathrm{tr}\left(\mathcal{L}_2 R - \mathcal{L}_1 + \frac{\rho}{2}\left(RXX^{\top}R^{\top} - 2XX^{\top}R^{\top} + 2EX^{\top}R^{\top} + RR^{\top} - ZR^{\top}\right)\right).
\end{aligned}$$

Next, the critical points are found by taking the derivative, resulting in

$$\rho R\left(XX^\top + I\right) + \mathcal{L}_2 - \mathcal{L}_1 X^\top + \rho\left(E - X\right)X^\top + \rho Z = 0. \tag{4.24}$$

And so the update step R^{k+1} is defined as

$$R^{k+1} = \left(\frac{1}{\rho}\left(\mathcal{L}_1 X^\top - \mathcal{L}_2\right) + Z + \left(X - E\right)X^\top\right)\left(XX^\top + I\right)^{-1}. \tag{4.25}$$

The two Lagrange multipliers $\mathcal{L}_1, \mathcal{L}_2$ are dual variables, and they monotonically accumulate in their update step. The update steps enforce the two constraints on the and so are

$$\mathcal{L}_1^{k+1} = \mathcal{L}_1^k + \rho\left(X - RX - E\right) \tag{4.26}$$

$$\mathcal{L}_2^{k+1} = \mathcal{L}_2^k + \rho\left(R - Z\right). \tag{4.27}$$

Algorithm 5 describes the robust LRA algorithm step-by-step. Its complexity is dominated by the singular-value thresholding in the Z update step. In this case, since R is a square $n \times n$ matrix, the runtime is $\mathcal{O}(n^3)$; however, since the algorithm processes each data set $X^{(i)}$ individually, the running time is dependent on the cardinality of the largest data set $\max\{|X^{(1)}|, \ldots, |X^{(k)}|\}$. Moreover, the running time of the algorithm can be greatly shortened by parallelizing the computation. The outer *for*-loop $X^{(i)} \in X$ can be mapped across systems and processed concurrently. The R update step can also be made faster by first calculating the Cholesky decomposition of the inverse term $\left(XX^\top + I\right)$. To use the Cholesky decomposition, the matrix must be symmetric and positive definite. Because of the quadratic term $XX^\top$ the matrix is necessarily symmetric and positive semi-definite. The addition of the identity term ensures all the singular values are greater than or equal to one, and so the matrix is also positive definite. The complexity of taking the decomposition is $\mathcal{O}(n^3)$, but the least squares problem that must be solved each iteration can be much faster.

Algorithm 5: Robust LRA Reconstruction Step

Input : $X = \text{diag}\left(\{X^{(i)} : i \in (1, \ldots, k)\}\right)$, the block diagonal data matrix,
μ, weight of the $l_{2,1}$ norm,
λ, weight of the low-rank norm, default 1.

Output: R, the k block diagonal reconstruction matrix, E, the error matrix.

begin
  for $X^{(i)} \in X$ **do**
    Initialize $R^{(i)}, Z, \mathcal{L}_1, \mathcal{L}_2 \leftarrow \mathbf{0}, \rho \leftarrow 1\text{e}^{-5}$.
    while *not converged* **do**
      $U, \Sigma, V^\top \leftarrow \text{SVT}\left(R^{(i)} + \rho^{-1}\mathcal{L}\right)$
      $Z \leftarrow U \ \text{diag}\left(\max\left(0, \Sigma - \lambda\right)\right) \ V^\top$
      $R^{(i)} \leftarrow \left(\rho^{-1}\left(\mathcal{L}_1 X^{(i)\top} - \mathcal{L}_2\right) + Z + \left(X - E\right) X^{(i)\top}\right)\left(X^{(i)} X^{(i)\top} + I\right)^{-1}$
      $E \leftarrow X^{(i)\top} - X^{(i)\top} R^{(i)\top} + \rho^{-1}\mathcal{L}_1^\top$
      for $j \in 1, \ldots, N_k$ **do**
        $\xi \leftarrow \left\|X^{(i)}[\cdot, j]\right\|$
        if $\xi > \mu$ **then**
          $E[:, j] \leftarrow ((\xi - \mu)/\xi) E[:, j]$
        end
        else
          $E[:, j] \leftarrow 0$
        end
      end
      $\mathcal{L}_1 \leftarrow \mathcal{L}_1 + \rho\left(X^{(i)} - R^{(i)} X^{(i)} - E\right)$
      $\mathcal{L}_2 \leftarrow \mathcal{L}_2 + \rho\left(R^{(i)} - Z\right)$
    end
  end
  $R \leftarrow \text{diag}\left(\{R^{(i)} : i \in (1, \ldots, k)\}\right)$
end

4.3 Experimental Results

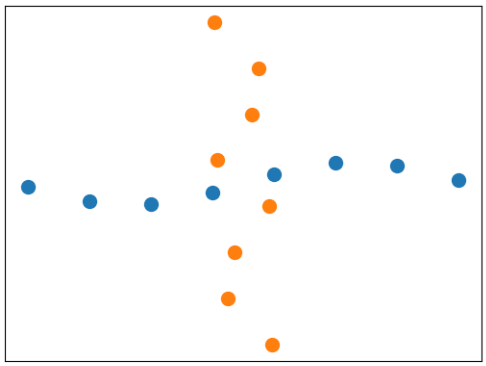
4.3.1 Synthetic Reconstruction Noise

To show the effectiveness of adding the sparsity term to LRA for a simple synthetic mixture of two manifolds X (a), the reconstruction matrix $R^{(X)}$ calculated by (b) locally linear embedding (LLE), (c) LRA, and (d) sparse LRA are compared in Figure 4.3. The samples (rows) of the data matrix X are ordered according to their manifold, where the blue horizontal manifold is listed first from left to right and the orange vertical manifold is listed second from top to bottom. The two manifolds are parameterized sine curves $y = (0.2)\sin(kx)$ with varying periods $k = 2, 20$.

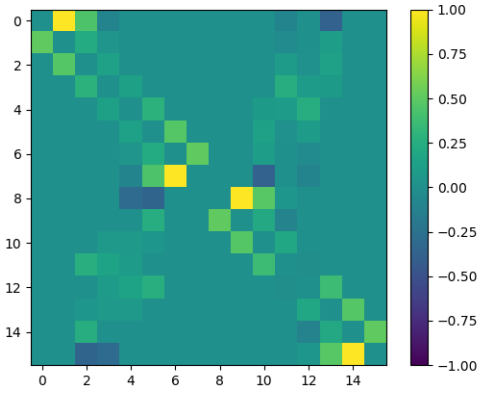
To construct $R^{(X)}$, LLE first calculates the nearest neighbors for each sample, in this experiment, five neighbors were used, then each sample is reconstructed from its neighbors in a barycentric manner by solving a system of linear equations. At the intersection of the two manifolds, LLE creates a number of strong short-circuits incorrectly fusing the manifolds together.

Rather than selecting exclusively neighboring points, the reconstruction matrix calculated by LRA is dense. The reconstruction values with high magnitudes are on the block diagonals, while only small magnitude noise values populate the off diagonal blocks. The block diagonal entries are correctly high, but to ideally separate the manifolds, the off diagonals entries should be zero.

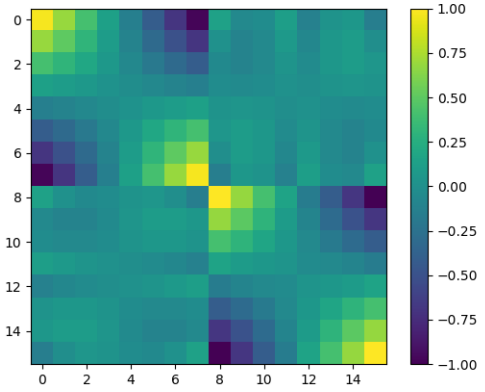
Like LLE, the reconstruction matrix SLRA produces is sparse, but more importantly, its sparsity is primarily on the off diagonal blocks. This means that SLRA is correctly eliminating the noisy connections between the manifolds in its reconstruction.



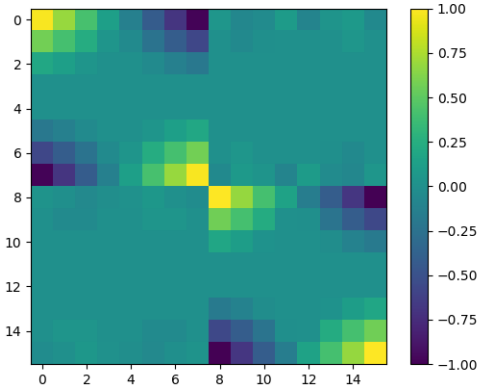
(a) Samples X drawn from a mixture of two manifolds.



(b) $R^{(X)}$ generated by locally linear embedding (LLE) with 5 neighbors.



(c) $R^{(X)}$ generated by low rank embedding (LRA).



(d) $R^{(X)}$ generated by sparse low rank embedding (SLRA).

Figure 4.3: A mixture of manifolds X (a) and the associated reconstruction matrix $R^{(X)}$ from LLE, LRA, and SLRA. The samples (rows) in $R^{(X)}$ are ordered according to the manifolds, with the blue points listed first from left to right, followed by the orange points listed from bottom to top. Ideally, the off-diagonal entries are zero, indicating there are no short-circuits between the two manifolds. LLE mixes together the manifolds at their junction, LRA has small noisy values in the off-diagonals, and SLRA does the best at differentiating the manifolds; however, it does not define the manifolds with neighborhoods like LLE.

4.3.2 Aligning Noisy News

To show the effectiveness of robust LRA on real-world data, the multi-lingual categorical Reuters data set from section 3.6.3 was used again. The data are news documents collected from Reuters from six popular categories in five languages: English, French, German, Italian, and Spanish [2]. The data were represented as a TF-IDF transformed bag of words model using the top 11-35K words were used for each language.

Two sets of experiments were performed. In each set, English language news articles were aligned with French, German, Italian, and Spanish articles, individually, making for four different alignment tests. In the first set of four experiments, the data were moderately corrupted with mean zero additive Gaussian noise, and in the second set, data were highly corrupted with a higher variance noise. For each experiment, 400 documents from English were aligned with the same documents in one of four foreign languages. For each alignment, 20% of the documents were randomly selected and provided to the algorithm as correspondences. After aligning and embedding the documents, a k-means algorithm was applied to make cluster predictions. Two metrics were used to evaluate the performance: adjusted mutual information (AMI) score and adjusted rand index (ARI) score. These metrics are described in section 3.6.3. Each experiment was repeated 20 times, with randomly selected correspondences in each iteration.

Results of all the noisy Reuters experiments are listed detailed in Table 4.1. In each test, RLRA outperformed or matched the performance of traditional LRA. As expected, the difference between the methods using the moderately corrupted data was smaller than their difference using the highly corrupted data. This suggests that there is a practical trade-off in using RLRA or LRA, between robustness to noise and an increased computation time required by the iterative algorithm.

Highly Corrupted Data				
	Adjusted Mutual Information Score			
	French	German	Italian	Spanish
LRA	0.130 ± 0.015	0.184 ± 0.020	0.128 ± 0.015	0.128 ± 0.019
R-LRA	0.173 ± 0.016	0.227 ± 0.009	0.166 ± 0.020	0.168 ± 0.017
	Adjusted Rand Index Score			
	French	German	Italian	Spanish
LRA	0.141 ± 0.016	0.197 ± 0.025	0.137 ± 0.014	0.110 ± 0.022
R-LRA	0.184 ± 0.018	0.243 ± 0.014	0.177 ± 0.012	0.152 ± 0.011
Moderately Corrupted Data				
	Adjusted Mutual Information Score			
	French	German	Italian	Spanish
LRA	0.284 ± 0.021	0.360 ± 0.018	0.308 ± 0.025	0.312 ± 0.012
R-LRA	0.334 ± 0.016	0.374 ± 0.014	0.336 ± 0.019	0.346 ± 0.014
	Adjusted Rand Index Score			
	French	German	Italian	Spanish
LRA	0.290 ± 0.017	0.345 ± 0.017	0.300 ∓ 0.018	0.306 ± 0.008
R-LRA	0.336 ± 0.013	0.359 ± 0.015	0.329 ± 0.015	0.335 ± 0.009

Table 4.1: Results from clustering a corrupted version of the Reuters cross-lingual document corpus, comparing low rank alignment (LRA) against robust LRA (RLRA). For both metrics, higher is better.

4.3.3 Calibration Transfer for Noisy Raman

To demonstrate the robustness of RLRA to sample-specific additive noise, the kind seen in Figure 4.1 (c) and (d), in this experiment, Raman spectra from multiple instrument recorded under varying conditions are aligned. This is another example of the task *calibration transfer*, first described in section 3.6; however, Raman spectroscopy is very different from the laser-induced breakdown spectroscopy (LIBS) of that experiment. Raman spectroscopy non-destructively analyzes matter by observing low-frequency light scatter interacting with the analyte. Because the technology can be made into a rugged portable unit, Raman instruments are used *in situ* for tasks as remote as mineral identification on the Martian surface [13, 66], and because of its gentle nature, it can be even used on live human tissue [14, 70].

The data set used was a suite of 96 pure mineral powders analyzed on 11 different Raman instruments using an array of geometries and laser energies [24]. The example spectra in Figure 4.4 show that differences in peak presence/absence, position, and relative intensity are evident in existing data sets. These differences can be the result of sample factors (e.g. grain size, transparency, crystallographic orientation, grain surface), instrument factors (e.g. laser wavelength, power, and spot size, spectrometer resolution and sensitivity), experimental factors (e.g. angle of incidence and takeoff and the use/absence of polarizers) and data gathering factors (e.g. integration time, averaging, method/frequency of calibration). The data were processed using standard protocols at each institution. These typically include white light, CCD dark-field, substrate and fluorescence corrections, and subtraction of cosmic ray events. The spectra were not baseline corrected, so they are guaranteed to contain additive baseline noise.

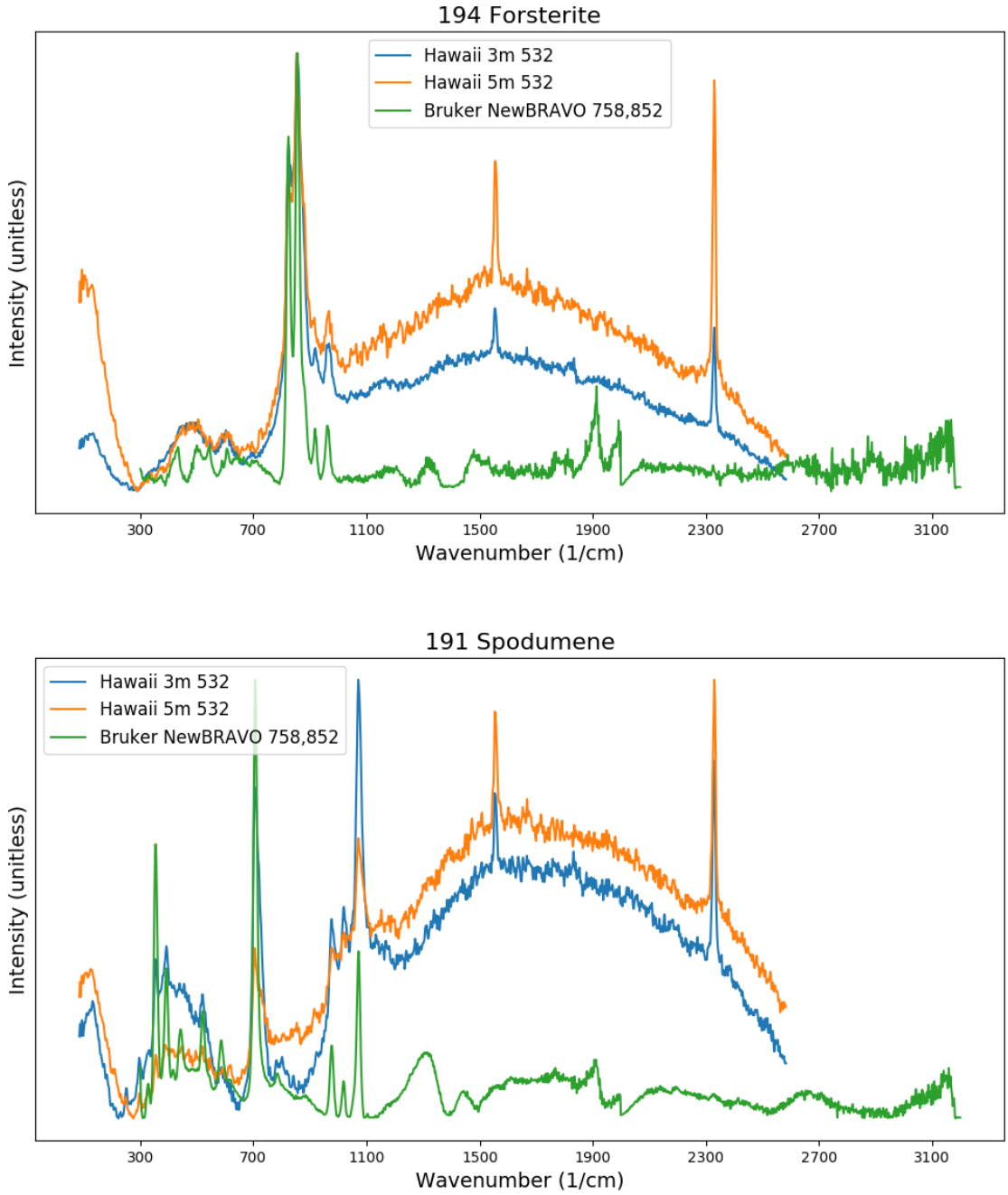


Figure 4.4: Raman spectra of two samples, *Forsterite* and *Spodumene*, from the from the Dyar96 data set. Each spectra is colored according to the instrument it was recorded on. The great different between instruments is clearly visible in both samples. Noise is also present in both samples, but it occurs to varying degree depending on instrument and sample.

In this experiment two instruments were selected, the *Bruker NewBRAVO 758,852* and the *Hawaii 532* (which is actually a Meade ETX-125 housed at the University of Hawaii). These two instruments were selected because the spectra they produce both have a large and distinct baseline continuum. The continuum is an unwanted signal, variously referred to as baseline, background, or simply noise by different communities depending on choice of the spectroscopic methods and the underlying causal physical processes. The baseline is typically additive noise that can effect each sample distinctly, based upon the composition of the sample [32,33]. Additionally, one instrument was used at two stand-off distances from the target, *Hawaii 3m 532* and *Hawaii 5m 532* at 3m and 5m respectively. In total, the data set consisted of three copies of the same 96 samples, but the features between instruments varied, where the *Bruker* instrument recorded on 1451 channels from 300 to 3200 wavenumber (1/cm) and the *Hawaii* instrument recorded on 1396 from 84 to 2583 wavenumber (1/cm).

Three experiments were conducted: (1) aligning the spectra from the two *Hawaii* instrument configurations, (2) aligning the spectra from the two instruments *Hawaii 3m 532* and the *Bruker*, and (3) aligning the spectra from *Hawaii 5m 532* and the *Bruker*. To compare the performance of RLRA on these tasks, it was again compared to manifold alignment, affine matching alignment, and Procrustes alignment. The alignments were limited to two instruments per experiment to allow for more competing methods, but it is worth noting that RLRA and manifold alignment are the only methods capable of simultaneously aligning more than two set. For each experiment, 10-fold cross validation was performed, where the algorithms were given the training data as correspondences. After embedding the training and testing samples, the embedded neighborhood of each held-out test sample was analyzed to see if it included its corresponding test sample from the other data set, i.e., are corresponding test samples x_1 and y_1 k-nearest neighbors. If all test sample embedding k-neighborhoods contain their corresponding sample, then the accuracy is 1, and if

none of the neighborhoods contain their corresponding samples, then the accuracy is 0.

The results of the experiments are listed in Table 4.2. Based upon the experimental setup, aligning the two *Hawaii* data sets should be an easier task because both data sets were recorded on the same instrument. This holds true in the results as well, where the neighborhood accuracy is approximately twice as high for each method compared to the other two experiments. In all three experiments, RLRA outperforms all of the competing methods, especially so when aligning the two stand-off distance *Hawaii* sets. This is likely due to smooth baseline error present in these two data sets. The baseline in the *Bruker* data set is noisier and less smooth than that found in either of the *Hawaii* data sets.

Setting $k = 10$ is a more lenient metric than $k = 5$, so the methods uniformly perform better. However, the superior performance of RLRA is more clearly indicated by its accuracy at $k = 5$. While the other methods evaluated had approximately a 40% drop in performance from $k = 5$ to 10, RLRA only performed about 18% worse with the stricter metric. This further validates that the embedding produced by RLRA very accurately aligns the disparate data sets.

Hawaii 5m 532 ↔ Hawaii 3m 532		
	Neighborhood Accuracy	
	$k = 10$	$k = 5$
Affine	$0.52 \pm .014$	$0.36 \pm .016$
Manifold	$0.44 \pm .014$	$0.28 \pm .008$
LRA	$\mathbf{0.83 \pm .012}$	$\mathbf{0.75 \pm .010}$
Procrustes	$0.52 \pm .011$	$0.34 \pm .012$
Hawaii 5m 532 ↔ Bruker NewBRAVO 758,852		
	Neighborhood Accuracy	
	$k = 10$	$k = 5$
Affine	0.24 ± 0.014	$0.16 \pm .014$
Manifold	0.23 ± 0.016	$0.14 \pm .012$
LRA	$\mathbf{0.39 \pm 0.014}$	$\mathbf{0.25 \pm .014}$
Procrustes	0.22 ± 0.006	$0.18 \pm .010$
Hawaii 3m 532 ↔ Bruker NewBRAVO 758,852		
	Neighborhood Accuracy	
	$k = 10$	$k = 5$
Affine	0.28 ± 0.009	$0.18 \pm .009$
Manifold	0.17 ± 0.009	$0.13 \pm .011$
LRA	$\mathbf{0.37 \pm 0.014}$	$\mathbf{0.31 \pm .016}$
Procrustes	0.34 ± 0.007	$0.18 \pm .012$

Table 4.2: Cross validation results for the Dyar96 Raman spectra alignment experiment. The neighborhood accuracy ranges from 0-1, worst to best, and is calculated based upon the number of test samples that contained their corresponding sample (in the other data set) within its k-nearest embedded neighbors.

4.4 Remarks

This chapter presented two robust reformulations of LRA, one designed for suppressing short-circuits between mixed manifolds and one designed for outliers and general noise. Unlike traditional LRA that has a closed form solution in the first step for the calculation of the reconstruction matrix, both of these algorithms are iteratively solved using the alternating direction method of multipliers. To prevent low-value short-circuits from joining entangled manifolds, an l_1 penalty is added to the reconstruction step. For general additive noise and outliers, the error term is directly model with an $l_{2,1}$ penalty, to allow for variation in error on a sample-by-sample basis. Although both of these additions increase the running time of the algorithm, they are experimentally shown to outperform traditional LRA in noisy settings.

CHAPTER 5

MIXED MANIFOLD DOMAIN ADAPTATION

This dissertation has so far described algorithms for aligning multiple data sets, where the data from each set lie on a mixture of lower-dimensional manifolds. The manifold alignment algorithm calculates an embedding for each of the heterogeneous data sets to a shared space. A few common usages for manifold alignment include visualizing high-dimensional data, finding similar samples and relationships between heterogeneous representations, and cluster analysis. However, these are all unsupervised tasks that do not incorporate label information. When the subsequent task post-alignment is supervised, like classification or regression, then the data are likely partially-labeled, and the alignment algorithm should ideally leverage these labels. If the data sets are assumed to share a label space, then this information can be used in conjunction with correspondences to better align the disparate data sets. While this may seem like a niche problem setting, this type of data is the norm rather than the exception in a majority of space science applications. In the case of ChemCam, there are data sets of LIBS spectra of rocks recorded in simulated Martian conditions from labs at Los Alamos National Laboratory, Mount Holyoke College, and CNES France that all share the same response surface, which is chemical composition in weight % oxide. In this chapter, a supervised version of LRA is presented that maximizes the correlation of the embedded samples with their label information while preserving the geometry within each data set and the relationships between sets.

5.1 Background

The number of large data sets available to machine learning practitioners is proliferating, and algorithms capable of using these big data sets have been shown to vastly outperform models trained with fewer examples. Unfortunately, the field of transfer learning has shown that it is difficult to improve the performance of a model by incorporating multiple disparate data sets unless careful preprocessing is applied to mitigate inter-set differences. Despite this obstacle, learning concurrently from many data sets provides the potential to improve accuracy and increase robustness. *Domain adaptation* (DA) is a sub-field of transfer learning that seeks to use one or more source data sets to assist in predicting a target data set that has been drawn from a different but related distribution. Heterogeneous domain adaptation is a more general case of DA, where the source and target sets are not required to share the same feature representation or dimensionality.

Unsupervised domain adaptation is the subfield of DA that seeks to solve problems where no label information is known for the target data set. These methods are widely studied in the literature [28, 38, 59, 74] and perform well when label information is not available. Unsupervised DA methods are often agnostic to the subsequent task (e.g., classification, regression). While this results in wide applicability, unsupervised DA cannot benefit from label information in cases where it is available.

Supervised and semi-supervised DA are subfields that seek to solve the problems where label information is present, if only partially, in the source and target domains. These methods are also widely studied [20, 83]; however, prior work has largely assumed that the label information is categorical and the final task is classification. In a regression setting with continuous labels, label preprocessing techniques, like binning or clustering, may be used to discretize the label information. These techniques enable the use of existing DA methods, but this work demonstrates that such label manipulation is an imperfect stopgap. Supervised methods that natively handle

continuous labels have been proposed [19, 29], but the label information is not used during adaptation time but rather during the task time.

Some previous work on DA for regression has been done, but most focuses on calculating sample weights for the regression algorithm. An SDP-based DA algorithm for learning sample weights for regression is presented in [16, 17], including pointwise learning guarantees. However, it is assumed that the source and target distributions be homogeneous and "reasonably close". Source and target set bias correction with continuous labels is discussed in [94] using a modified version of [19] followed by sample weight calculations, but this too requires a homogeneous feature space representation across data sets. Lastly, a correlation based approach to domain adaptation is presented in [8], but instead of maximizing the correlation between feature and label spaces, they focus on finding correlated features between source and target.

The problem setup of continuous label information where the label features are matching across data sets was inspired by our work with the science team of the ChemCam instrument aboard the Mars rover *Curiosity*, which analyzes rocks and soils using a laser-induced breakdown spectrometer (LIBS). Although the rover itself has ten calibration samples of known composition on board, the breadth of compositional diversity on the Martian surface requires a far broader calibration suite to interpret chemical compositions from instrument data. To this end, data from terrestrial laboratories are being used to build prediction models that can subsequently be applied to Mars data. Although surface pressures and temperatures on Mars can be simulated in the laboratory, and the flight model sent to Mars was used to build a small calibration suite of 69 samples before launch [91], the calibration suites in current use suffer from two major shortcomings. The first is their small size relative to the target data,and the second stems from a wide variety of collection parameters, with samples recorded by different instruments under many operating conditions, including variations in laser power, distance from target, and gravitational field strength. More-

over, multiple laboratories, including the Mineral Spectroscopy Laboratory at Mount Holyoke College, are collecting compositional data on standard geological samples. Thus, the challenge is to align the data sets from these disparate terrestrial instruments, in combination with the small amount of data on the ten calibration targets on Mars, and use the resulting composite suite to interpret data on the unknown rock and soil targets from Mars. Each data set in this spectroscopy setting has the same continuous label information, namely the weight percent of oxides like SiO_2, which means that the task of interest for composition prediction is regression.

This motivates a DA method that maximizes correlation within each data set and between data sets, while maintaining the geometry intrinsic to each data set, for the subsequent task of regression. Correlation analysis for domain adaptation (CADA) solves this problem simply and quickly by mapping the sets to a lower-dimensional joint space. CADA formulations are presented here for linear and non-linear maps, for both the primal and dual problems. CADA is especially well-suited to data from many types of scientific instrumentation, where it is often necessary to correct for differences arising from variable experimental geometries (close-up vs. long distance measurements), environmental conditions (e.g., deep sea vs. ambient lab conditions vs. the Martian surface), and analytical parameters such as laser wavelength, power density, and beam size.

5.2 Correlation Analysis for HDA

The algorithm presented here calculates a low-dimensional mapping that maximizes the inter-set correlations while preserving the intra-set geometries. So, it is necessarily related to canonical correlation analysis (CCA). However, unlike CCA, this algorithm is designed to solve the problem of domain adaptation of multiple data sets of differing samples. Furthermore, the method aligns the data sets while maintaining their individual mixed manifold geometries. A linear version of the algorithm

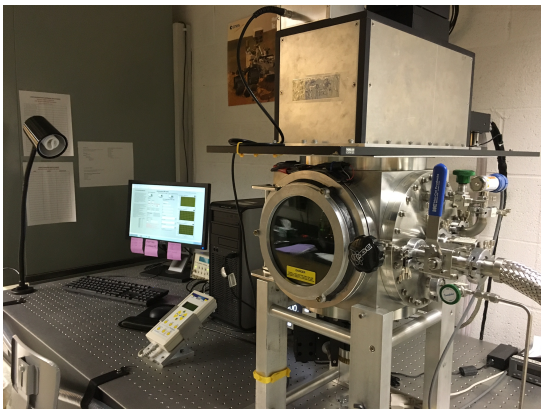


Figure 5.1: Laser-induced breakdown spectroscopy (LIBS) instrumentation used to record the spectra in the Mineral Spectroscopy Laboratory at Mount Holyoke College. The LIBS laser pulses the mineral sample in a nearly evacuated chamber under a CO_2 atmosphere to create a plasma. Using mirrors, the light emitted from the plasma is passed through a diffraction grating to separate the beam into three frequency ranges. The three sub-beams are directed to three charge-coupled devices (CCD), which are sensitive to different frequencies. The number of photons that strike the surface of each CCD is recorded to produce a spectrum.

is presented first, followed by a kernelized version that allows for the calculation of non-linear maps. In the case of scientific instrument data, it is typical to have the number of features greatly outnumber the number of samples, so it is often more efficient to use the latter kernel dual formulation.

Problem Description

Given k data sets consisting of heterogeneous feature matrices $X_1, \ldots, X_k$ where $X_i \in \mathbb{R}^{n_i \times p_i}$ and corresponding homogeneous response matrices $Y_1, \ldots, Y_k$ where $Y_i \in \mathbb{R}^{n_i \times q}$, the purpose of the algorithm is to find functions $f_1, \ldots, f_k$ that map the disparate data sets into a joint space $f_i : \mathbb{R}^{p_i} \mapsto \mathbb{R}^d$ such that $d \leq p_i$ for $i = 1, \ldots, k$. The goals of the mappings are to preserve the feature space geometry of each data set while using the response information to preserve the response surface geometry between data sets. No correspondences between the data sets are given, but all of the response values $Y_1, \ldots, Y_k$ are assumed to be measurements of the same attribute.

No assumptions are made about the similarity of the feature matrices. When $d \leq 3$, this algorithm may be used for data visualization, but this chapter will instead focus on the task of *domain adaptation* [20].

Without loss of generality, denote (X_1, Y_1) as the *target* data set and the remaining $(X_2, Y_2), \ldots, (X_k, Y_k)$ as *source* data sets. In this task, the goal of the algorithm is to map the source and target sets to a joint space so that a regression model predicting the target set can be trained using all of the source sets as well.

Linear formulation

The CADA algorithm treats each input data set as a mixture of manifolds. It calculates the joint space maps $f_1, \ldots, f_k$ by maximizing the correlation between the varying data source representations $X_1, \ldots, X_k$ and their shared response surface $Y_1, \ldots, Y_k$ while preserving the geometry of each data set.

To begin the algorithm, define the joint data, response, and mapping matrices $X \in \mathbb{R}^{N \times P}$, $Y \in \mathbb{R}^{N \times q}$, and $f \in \mathbb{R}^{P \times k(d)}$, respectively, as

$$X = \begin{bmatrix} X_1 & & 0 \\ & \ddots & \\ 0 & & X_k \end{bmatrix}, Y = \begin{bmatrix} Y_1 \\ \vdots \\ Y_k \end{bmatrix}, f = \begin{bmatrix} f_1 \\ \vdots \\ f_k \end{bmatrix}$$

where $N = \sum_{i=1}^{k} n_i$ and $P = \sum_{i=1}^{k} p_i$. By stacking the individual response matrices Y_i, the algorithm will be able to reason across data sets. To simplify the algorithm's description, it is assumed that the response matrix Y has been mean-centered.

The CADA algorithm optimizes the following objective function

$$\underset{f}{\text{maximize}} \;\; \frac{1}{2} \left\| \text{Corr}(Xf, Y) \right\|^2 + \alpha \text{Geo}(Xf, R) \text{ s.t. } \left\| CXf \right\|^2 = 1 \tag{5.1}$$

where $\text{Corr}(\cdot)$ is the sample correlation penalty term, $\text{Geo}(\cdot)$ is the geometric penalty term, α is a non-negative mixing parameter controlling the trade-off between maxi-

mizing correlation and preserving geometry, and R is the reconstruction matrix described later. The geometric penalty term preserves the mixed manifold structure of the individual data sets, while the correlation penalty reasons between data sets by finding a mapping that best prepares the data for the subsequent task of regression. These terms are now described in more detail.

Geometric penalty

The geometric penalty term preserves individual data set geometries by modeling each as a mixture of manifolds during domain alignment [9]. To calculate the mixture of manifolds representation for each data set, a low-rank reconstruction matrix $R^{(X_i)} \in \mathbb{R}^{n_i \times n_i}$ must be calculated for each data set as

$$\underset{R^{(X_i)}}{\text{minimize}} \;\; \frac{1}{2} \left\| X_i - R^{(X_i)} X_i \right\|_F^2 + \left\| R^{(X_i)} \right\|_* ,$$

where $\|\cdot\|_*$ is the nuclear norm. This can be solved in closed form using singular value decomposition (SVD) [27]. First, X_i is decomposed using SVD, $X_i = USV^\top$. Next, the columns of V and S are partitioned into $V = [V_1 V_2]$ and $S = [S_1 S_2]$ according to the sets

$$I_1 = \{i : s_i > 1 \;\; \forall s_i \in S\} \text{ and } I_2 = \{i : s_i \leq 1 \;\; \forall s_i \in S\}.$$

Then the reconstruction matrix $R^{(X_i)}$ is calculated as

$$R^{(X_i)} = V_1 (I - S_1^{-2}) V_1^\top . \tag{5.2}$$

Afterward calculating the reconstruction matrices, the block diagonal matrix $R \in \mathbb{R}^{N \times N}$ is constructed similarly to X by stacking the matrices $R^{(X_1)}, \ldots, R^{(X_k)}$.

The geometric penalty term can now be defined as

$$\text{Geo}(Xf, R) = -\frac{1}{2} \|Xf - RXf\|^2 . \tag{5.3}$$

Intuitively, the term can be understood as preserving the reconstruction matrix from the original space in the low-dimensional joint space. The penalty is negative because the CADA objective function, equation 5.1, is maximized in the optimization.

Correlation penalty

To maximize the correlation between the mapped data Xf and the centered response data Y, the sample covariance is calculated as

$$\text{Cov}(Xf, Y) = \sum_{a=1}^{k} \frac{1}{n_a} \sum_{i=1}^{n_a} (x_a^i f_a - \overline{x_a^j f_a})^\top (y_a^i),$$

where $\overline{x_a^j f_a}$ is the sample mean. To simplify this equation, the expectation can be eliminated by mean-centering the mapped data,

$$\begin{aligned} \text{Cov}(Xf, Y) &= \frac{1}{N} \sum_{a=1}^{k} \sum_{i=1}^{n_a} (C_a x_a^i f_a)^\top (y_a^i) \\ &= \frac{1}{N} f^\top X^\top C Y, \end{aligned}$$

where $C_a \in \mathbb{R}^{n_a \times n_a}$ is the *centering matrix* $C_a = I - (1/n_a)\mathbf{1}\mathbf{1}^\top$ and $C \in R^{N \times N}$ is the block diagonal matrix composed of centering matrices.

To calculate the correlation, the covariance must be scaled by the standard deviation, yielding the joint space sample correlation

$$\text{Corr}(Xf, Y) = \frac{f^\top X^\top C Y}{\|CXf\| \, \|Y\|}. \tag{5.4}$$

Analysis of the Algorithm

It is shown here that the CADA objective function can be solved in closed form using as a single generalized eigenvalue problem. Afterward, the complexity of the algorithm is discussed.

Theorem 2. *The function f that minimizes the CADA objective function in equation 5.1 is given by the eigenvectors corresponding to the largest eigenvalues of the generalized eigenvalue problem*

$$\frac{1}{\|Y\|^2} X^\top \left(CYY^\top C - \alpha \|Y\|^2 M \right) Xf = X^\top CXf,$$

where $M = (I - R)^\top (I - R)$ for the identity matrix I.

Proof. The correlation term in equation 5.1 is first discussed. From equation 5.4, it follows that

$$\begin{aligned} \|\mathrm{Corr}(Xf, Y)\|^2 &= \frac{\left\| f^\top X^\top CY \right\|^2}{\|CXf\|^2 \|Y\|^2} \\ &= \frac{Tr\left(f^\top X^\top CYY^\top CXf \right)}{\|Y\|^2 Tr\left(f^\top X^\top CXf \right)}. \end{aligned}$$

To optimize this equation, the constraint $\|CXf\|^2 = 1$ is enforced using the Lagrange multiplier λ,

$$\frac{1}{\|Y\|^2} Tr\left(f^\top X^\top CYY^\top CXf \right) - \lambda Tr\left(f^\top X^\top CXf \right).$$

The constraint here does not affect the correlation, but rather regularizes the mappings f.

Next, the geometric penalty is considered. From equation 5.3, it follows

$$\begin{aligned}\text{Geo}(Xf, R) &= -\frac{1}{2}Tr((Xf - RXf)^\top (Xf - RXf)) \\ &= -\frac{1}{2}Tr(f^\top X^\top (I - R)^\top (I - R)Xf) \\ &= -\frac{1}{2}Tr(f^\top X^\top MXf)\end{aligned}$$

Combining the correlation and geometric penalties forms the equation,

$$\frac{1}{\|Y\|^2}Tr\left(f^\top X^\top \left(CYY^\top C - \alpha \|Y\|^2 M\right) Xf\right) - \lambda Tr\left(f^\top X^\top CXf\right). \tag{5.5}$$

Using standard methods from calculus, the optimal value of equation 5.5 is given by the generalized eigenvalue problem,

$$\frac{1}{\|Y\|^2}X^\top \left(CYY^\top C - \alpha \|Y\|^2 M\right) Xf = \lambda X^\top CXf. \tag{5.6}$$

The eigenvectors are the d columns of f and so the mapping f_1, for example, is the first p_1 rows of f. Therefore, equation 5.1 is maximized by using the eigenvectors associated with the d largest eigenvalues. □

The CADA algorithm is written out in algorithm 6. In step 1, the calculation of the individual reconstruction matrix $R^{(X_1)}, \ldots, R^{(X_k)}$ can be easily parallelized across the k data sets. The dominating factor for the runtime cost of the algorithm is the final $N \times N$ generalized eigenvalue problem. However, this can be calculated efficiently because all of the matrices involved are symmetric and sparse.

To make α easier to tune, it can be beneficial to first scale terms such that $\left\|CYY^\top C\right\| = 1$ and $\|M\| = 1$.

In addition to mapping data to low-dimensional space, CADA can also be used to transfer data from one data set representation to another by using the joint space

Algorithm 6: Correlation Analysis for Domain Adaptation (CADA)

Input: block diagonal data matrix X,
centered block stack label matrix Y,
dimension d, and weight μ.
Output: mapping matrix f.
Step 1: Compute the geometric reconstruction matrix R by calculating the matrices $R^{(X_1)}, \ldots, R^{(X_k)}$ according to equation 5.2.
Step 2: Set f equal to the d eigenvectors associated with the largest eigenvalues in equation 5.6.

as a intermediary. For example, to view X_1 in the X_2 representation one would use the mapping $f_1 f_2^{\dagger}$ where $f_2^{\dagger}$ is the pseudo-inverse. CADA may also be used as a preprocessing step before other domain adaptation methods.

5.3 Kernel Formulation

For some problems, a linear mapping will not suffice to align the disparate data sets well. The CADA algorithm can be modified in the style of Laplacian eigenmaps [5] to provide non-linear embeddings of the given data sets, but this does not provide a natural out-of-sample extension, which is critical to the ultimate regression task. Instead, a kernelized version of CADA is described here that yields non-linear maps with an appropriate choice of kernel functions.

To begin to kernelize equation 5.6, all data sets X_i must first be replaced with their kernel mapping $\phi_i(X_i)$, where ϕ_i is a map to a possibly infinite-dimensional Hilbert space. Let Φ be the block diagonal matrix composed of entries $\phi(X_1), \ldots, \phi(X_k)$. After mapping the feature vectors to Hilbert spaces, the corresponding eigenvectors f must also be updated, yielding

$$\Phi^{\top} \left(CYY^{\top}C - \alpha M \right) \Phi g = \|Y\|^2 \, \Phi^{\top} C \Phi g,$$

where g are possibly infinite-dimensional eigenvectors. According to the Riesz representation theorem, the eigenvectors g can be represented as a linear combination of

mapped samples. Using the substitution $g = \Phi^\top h$ and left multiplying by Φ yields

$$\Phi\Phi^\top \left(CYY^\top C - \alpha M\right) \Phi\Phi^\top h = \|Y\|^2 \Phi\Phi^\top C\Phi\Phi^\top h.$$

The quadratic form $\Phi\Phi^\top$ can then be replaced with the kernel matrix K,

$$K \left(CYY^\top C - \alpha M\right) Kh = \|Y\|^2 KCKh,$$

where h is now an $N \times d$ matrix and the block diagonal kernel matrix K is $N \times N$.

In addition to providing a non-linear mapping, a linear kernel $k(X_i) = X_i X_i^\top$ may be used to convert linear CADA from its usual primal form to its dual form.

Scientific instrument data often have many more features P than samples N, i.e., $P \gg N$. In this situation, Kernelized CADA is better conditioned than linear CADA; however, with very large P, the norm of the joint maps $\|f\|$ tends to get large. To remedy potential over-fitting, a ridge penalty can be incorporated into the eigenvalue problem,

$$K \left(CYY^\top C - \alpha M\right) Kh = \left(\|Y\|^2 KCK + \beta I\right) h,$$

where the scalar β is the non-negative ridge penalty parameter. In practice, reasonable values for β are from 1E^{-5} to around 1, where larger values will shrink the size of $\|f\|$ more.

5.4 Experimental Results

To evaluate the performance of CADA, its performance was compared against the following competing DA methods: canonical correlation analysis (CCA) [44], partial least squares (PLS) [80], subspace alignment (SA) [28], heterogeneous domain adaptation (HDA) [83], domain adaptation for regression (DAR) [17], and domain adaptation for structured regression (DASR) [94].

CCA was used in two different configurations, CCA-Multi and CCA-Joint. In CCA-Multi, a different CCA model was fit on each of the data sets X_i, Y_i, where each model used the same d number of components. In CCA-Joint, a single CCA model was fit on the target set X_t, Y_t and then all other data sets were mapped to the same space using the model. CCA-Joint is only applicable when the source and target data sets share the same features.In final experiment, CCA failed to yield viable components, so similarly defined PLS-Multi and PLS-Joint were used instead.

The method most related to CADA is HDA, but a direct comparison is impossible because it assumes categorical labeled data. For the purpose of comparison, the continuous response values in each experiment are binned or clustered to produce a discrete set of labels to use with HDA.

A complete Python implementation of CADA is available for download from the author's web site, including demonstration code [1].

5.4.1 Oblate Spheroid Alignment

As a proof of concept, consider applying domain adaptation to a pair of oblate spheroid shells,

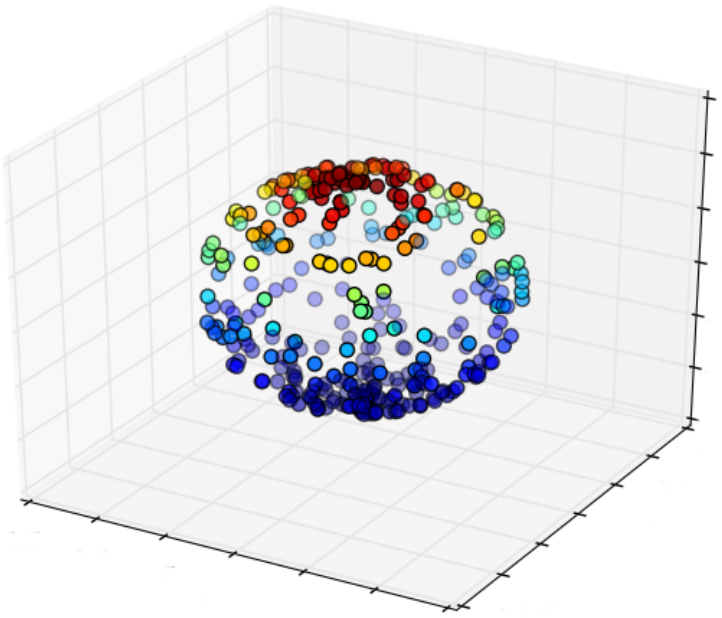
$$
\begin{aligned}
x &:= \sigma \cos(u) \sin(v) \\
y &:= \sigma \sin(u) \sin(v) \\
z &:= \cos(v)
\end{aligned}
$$

for parameters $u \in [0, 2\pi)$, $v \in [0, \pi]$, and a parameter σ that adjusts how "squashed" the sphere appears. In this first experiment, two spheroid shells were generated with $\sigma = 1, 3$ and uniform random sampling each u, v pair. The more oblate sphere was also tilted $\pi/4$ radians about the x-axis.

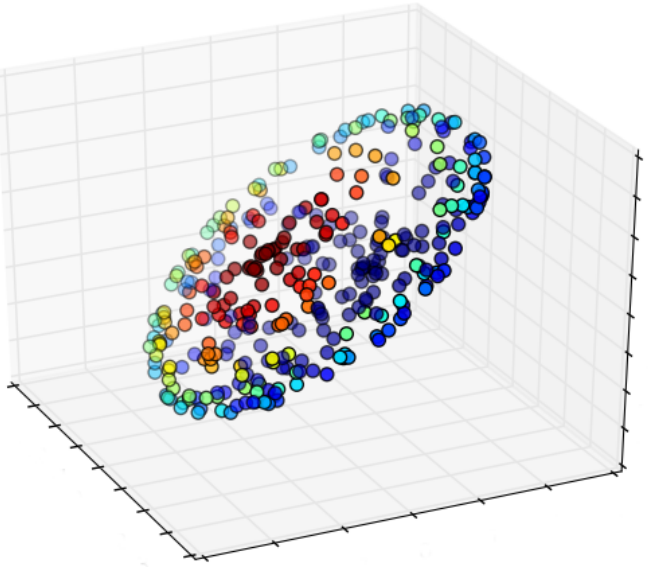
[1] https://github.com/all-umass/cada

To each spheroid, a simulated heat source is applied to a point on the surface, resulting in smooth diffusion of heat around the entire shell. The three-dimensional coordinates of each spheroid shell form data sets X_s and X_t, and the corresponding one-dimensional heat values are used as response matrices Y_s and Y_t. The resulting spheroid shells and their associated temperatures are shown in Figure 5.2.

All methods map the data to a 2-dimensional representation first, except for the methods DAR and DASR that do not reduce the dimensionality of the data. Figure 5.3 demonstrates the excellent performance of the proposed method, CADA. Traditional subspace alignment techniques do not take advantage of the available temperature information, and fail to produce an embedding conducive to the regression task. The CCA-based methods capture the connection between coordinates and temperature, but are limited by their omission of inter-data set similarity information. The binned HDA method produces reasonable results, but incurs the additional complexity of computing a fixed number of temperature bins. In the general case, this binning strategy is further limited by the dimensionality of Y_i: as dimension increases, the representative power of discrete bins falls dramatically. In addition to bin size, HDA has three tuning parameters, whereas all the other tested methods only have one parameter to tune. Only the CADA method fully exploits the relationships between and within data sets as well as the real-valued nature of the temperature regression targets, resulting in the most accurate regression performance among all tested methods. The DAR and DASR methods produce comparable results to CADA, achieving an MSE of 57.237 and 57.109 respectively. These two methods do not reduce the dimensionality of the data, which is likely to account for their strong performance since the data inherently lie in three-dimensional space.



(a) *Source* spheroid samples



(b) *Target* spheroid samples

Figure 5.2: Uniform random sampling of 50 and 40 points from the source and target spheroids, respectively. The points are colored according to their simulated temperature, where red is hotter and blue colder.

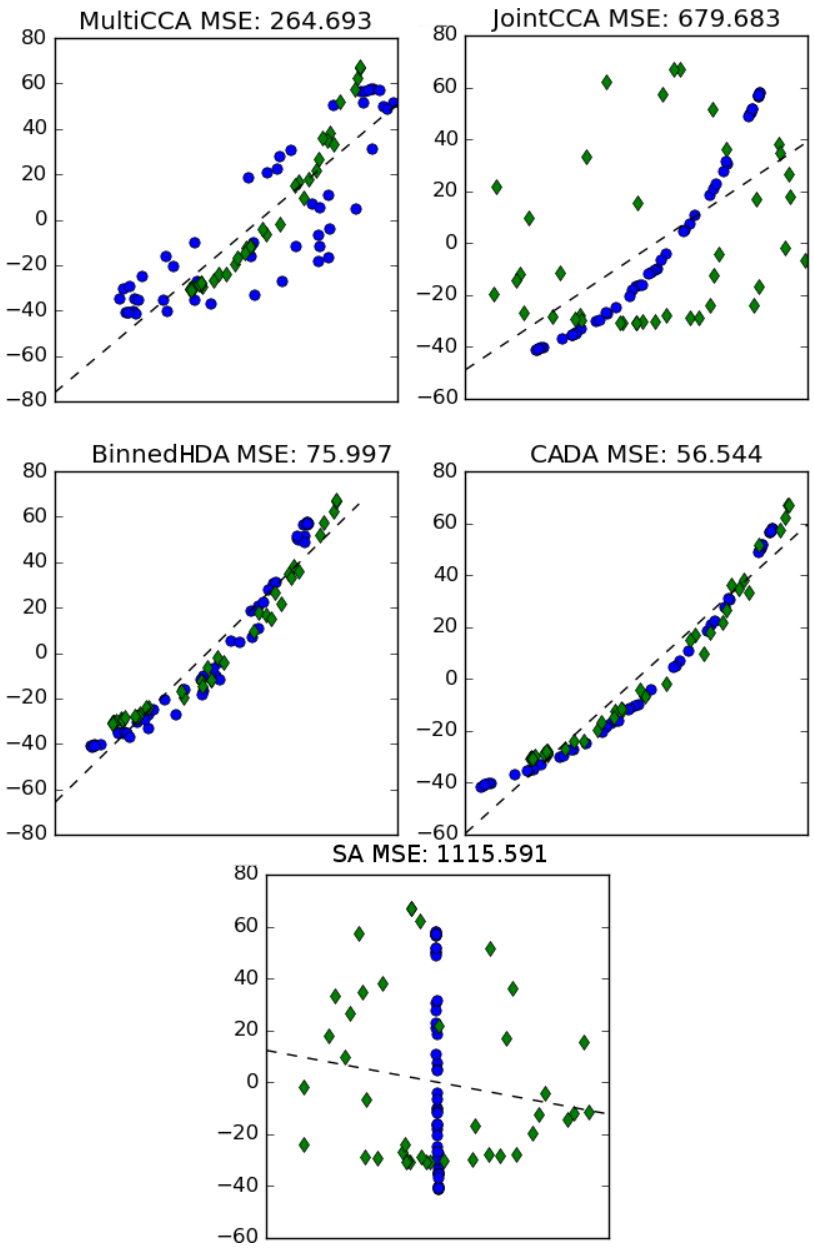


Figure 5.3: Comparison of 1-dimensional embedding from six competing DA methods: MultiCCA, JointCCA, a binned version of heterogeneous domain adaptation (BinnedHDA), correlation analysis for domain adaptation (CADA), and subspace alignment (SA). The blue circles are samples from the source set and the green diamonds are samples from the target set. The x-axis is the 1-D embedding and the y-axis is the temperature. A linear regression model was fit on temperature and is annotated as a dashed line. The mean squared error (MSE) for each model is reported at the top of each sub-figure. Temperature is generated by a univariate equation and so should naturally reduce to a 1-D representation; however, the generating function is non-linear, so these linear DA methods are unable to fully fit the curve.

5.4.2 WiFi Localization

To demonstrate the effectiveness of the proposed method in a real-world setting, we apply domain adaptation to data sets generated for the 2007 IEEE ICDM Data Mining contest [95]. The contest consisted of two tasks, each with a distinct set of WiFi signal strength readings and associated spatial coordinates. Each reading is a sparse vector of integer values that correspond to the signal strength (RSS) observed when connecting to the same number of wireless access points in unknown locations. The target set was 101-dimensional and the source set was sampled at half the rate, yielding a 50-dimensional representation. These readings are labeled with two-dimensional coordinates based on the position of the wireless receiver as it moved throughout the hallways of a building. The data for each task were collected in different time periods, so domain adaptation is suitable for transferring localization information over time.

For this experiment, the readings and coordinates of the competition's test sets are used to produces two supervised data sets for domain adaptation, with 2,137 entries for X_1 and 3,128 entries for X_2, each with an associated Y_i with the same number of rows. Several domain adaptation methods are tested by learning embeddings that map each X_i to a two-dimensional joint space, then evaluate the new representation's localization accuracy using linear regression.

The source and target sets do not share the same dimensionality, so only the heterogeneous DA methods were testable: BinnedHDA, MultiCCA, CADA, and Kernel-CADA. Results of this experiment, shown in Figure 5.4, demonstrate the effectiveness of the proposed method in both the linear and kernelized variants over a large range of joint space dimensionalities. BinnedHDA nearly matched the performance of linear CADA, but was much more sensitive to the joint space dimension and required a binning preprocessing step beforehand, where the k-means clustering algorithm was used to produce 20 spatially-grouped classes. Most notably, the kernelized CADA method

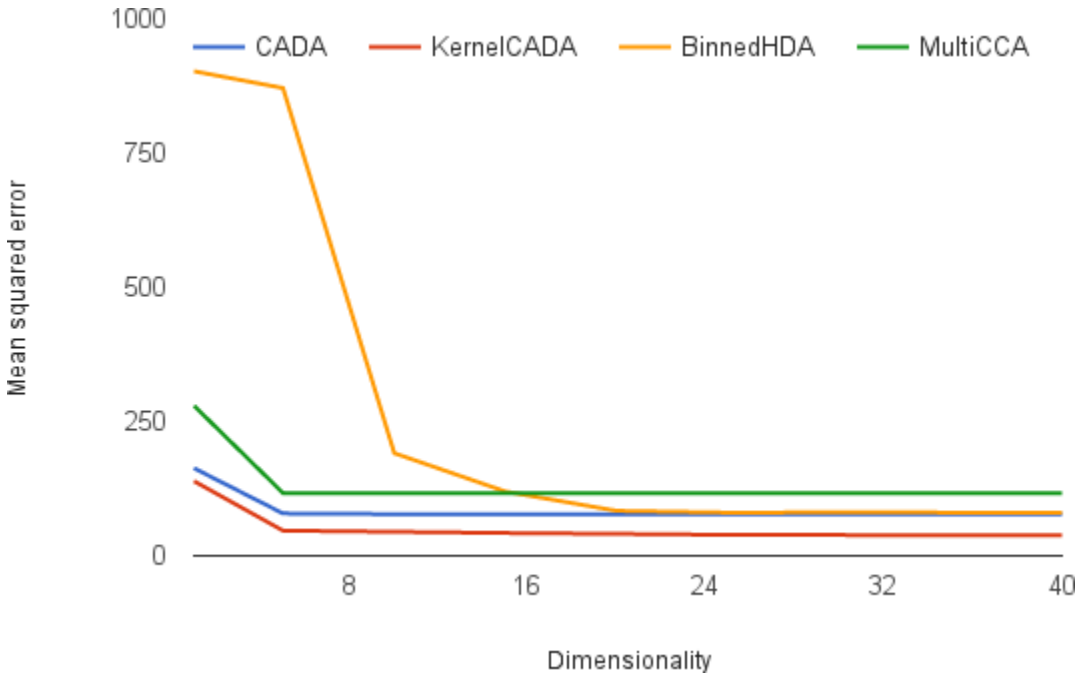


Figure 5.4: Comparison of regression error from four competing heterogeneous DA methods: MultiCCA, a binned version of heterogeneous domain adaptation (Binned-HDA), and two variants of correlation analysis for domain adaptation (CADA). The x-axis is the dimension of the joint space. The y-axis is the mean squared error (MSE) for a linear regression fit on each model's joint space representation to predict the 2-D spatial coordinate labels.

performed exceptionally well using a cosine similarity kernel.These additional parameters make kernel CADA slightly more complicated to tune, but result in appreciable gains in the subsequent regression task.

5.4.3 Calibration Transfer for Laser-Induced Breakdown Spectroscopy

In all spectroscopic applications, there is a need to ensure that differences in instrument, environment, and experimental conditions are mitigated. A *calibration curve* is the regression model used by an instrument to predict a response, often chemical composition, for a given spectrum. Calibration transfer (CT) is a technique for transferring a calibration curve from one instrument to another using a calculated transfer function, without the need to re-sample the calibration samples [30]. CT can also be used to transfer the calibration curve of an instrument from one set of environmental conditions to a differing set of conditions. CT provides an excellent solution to the task of reconciling data for inter- and intra-lab comparisons on Earth and in extraterrestrial applications.

From a machine learning perspective, CT is just another form of domain adaptation. By mapping all intra-instrument spectra to a joint space, a calibration curve can be fit using all available data. In this experiment, two data sets were recorded on two different laser induced breakdown spectroscopy (LIBS) instruments in support of NASA's Mars Science Laboratory team. The first set, *LANL*, was composed of 400 rock spectra recorded on 6144 wavelength channels under Mars-like conditions at Los Alamos National Laboratory using an instrument simulating the ChemCam instrument on the *Curiosity* rover. The second set, *MHC*, was composed of 280 rock spectra recorded on 6144 channels under similar conditions at Mount Holyoke College on a LIBS instrument manufactured by a different vendor and containing a different type of laser. Examples of the spectra are given in Figure 5.5. Over 300,000

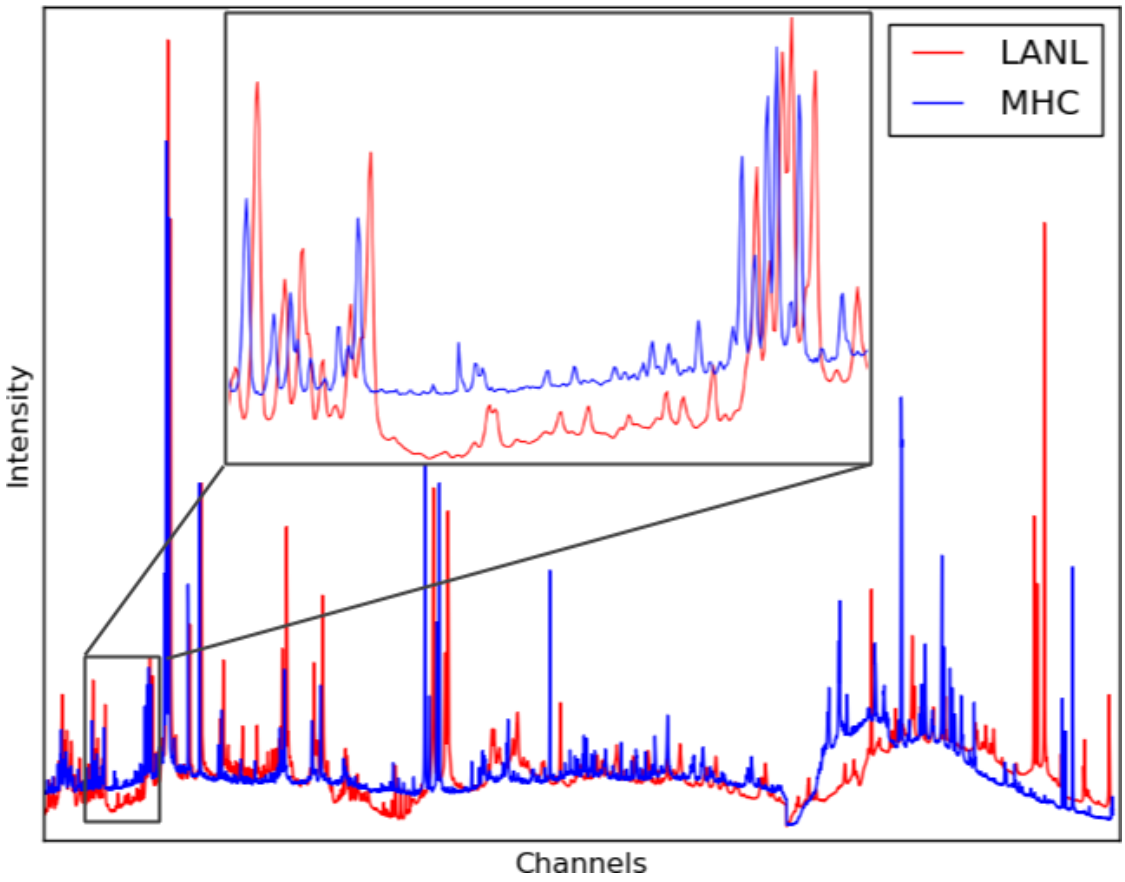


Figure 5.5: Mean LIBS spectra of mineral samples recorded at Los Alamos National Laboratory (LANL) and Mount Holyoke College (MHC). Instrumentation differences and varying experimental conditions induce discrepancies between the two sets of spectra, like the channel offset evident in the zoomed insert.

spectra have been recorded by the ChemCam instrument on Mars, but the chemical composition of those targets are currently predicted using only terrestrial laboratory calibration data. The spectra are used to measure the weight % oxide of various chemical components, including silica (SiO_2).

Spectra of the two data sets were first row-normalized by spectrometer following the preprocessing procedure described by [91]. After this step, the following cross validation procedure was repeated 25 times (with random shuffling of folds) for each of the DA methods compared: (1) the DA method was trained with an SiO_2 labeled training subset of the MHC and LANL spectra, (2) the DA method was then used to map the training set to the joint space, (3) a partial least squares (PLS) regression model was trained on the joint space training set, (4) the DA method was then used to map the testing set to joint space, (5) and lastly a PLS regression model was used to predict the testing SiO_2 concentration.

For each of the DA methods, the dimension of the subspace was tuned by searching $d \in [1, 200]$ over the training set. For RBF-CADA, the kernel width γ was set to the inverse of the median pairwise distance between training samples, and the ridge parameter β was tuned by searching over orders of magnitude from 1E^{-5} to 1.

Results of the experiment are listed in Table 5.1. RBF-CADA outperformed all other evaluated methods, better predicting the LANL set, the MHC set, and, consequently, the union of the two sets. The linear version of CADA was the second best performing method predicting the LANL set and the union, but the MHC target model proved to be more effective at predicting natively. Because of the poor problem conditioning, CCA-based methods and BinnedHDA provided no viable solution to this problem.

	LANL MSE	MHC MSE	All MSE
Source	255.9	868.7	NA
Target	39.9	27.9	NA
PLS-Multi	25.5	54.8	37.5
PLS-Joint	57.0	32.4	46.7
SA	43.0	48.1	45.3
DAR	43.1	43.0	NA
DASR	43.0	37.2	40.4
CADA	22.5	30.8	26.1
RBF-CADA	**22.0**	**27.5**	**24.4**

Table 5.1: Mean squared error (MSE) of prediction of SiO_2 for cross validation for each method evaluated over the Los Alamos National Laboratory (LANL) test set, Mount Holyoke College (MHC) test set, and both test sets combined.

5.5 Remarks

This chapter presented an algorithm for aligning mixtures of manifolds in the presence of label data, a supervised version of LRA customized for the downstream task of classification or regression. The data sets are assumed to share a label space, and this information is used in conjunction with correspondences to better align the disparate data sets. This was shown to be effective at transferring information from large labeled data sets to aid in the regression of smaller data sets. Unlike traditional LRA that cannot inherently embed out-of-sample data, the algorithm in this chapter directly calculates a mapping for out-of-sample extensions. Both a linear and a kernelized non-linear version of the algorithm were described.

CHAPTER 6

CONCLUSION

As manifold learning techniques are applied to more complex tasks, the problem of mixtures of manifolds has become increasingly apparent. In this dissertation, a new class of transfer learning algorithms is introduced for high-dimensional data sets that intrinsically lie on multiple low-dimensional manifolds. The proposed Low Rank Alignment framework is the first such alignment method to gracefully handle arbitrary mixtures of manifolds, a benefit that is reflected in the task performance comparisons. Two transfer learning problems are the primary focus within this dissertation, manifold alignment and heterogeneous domain adaptation.

First, the general LRA framework for aligning and embedding data sets is introduced. It uses a low-rank approximation instead of a nearest-neighbor graph for sample reconstruction, which is shown to be effective at avoiding short-circuits between entangled manifolds. Versions for both linear and non-linear manifolds are presented. Additionally, a small modification is introduced to improve its ability to perform clustering in the embedding space. Lastly, a method for actively selecting the most beneficially correspondences is described.

The next contribution of this dissertation is a robust extension of the LRA framework. Two versions of the algorithm are detailed, one that prevents low-value short-circuits from joining entangled manifolds, and one designed to deal with outliers and general additive noise by directly modeling the error. This addition forces the algorithm to be solved iteratively, but it is shown to outperform traditional LRA in noisy settings.

The final set of algorithms presented is a supervised framework for LRA for the transfer learning task of domain adaptation. The label data provide another means, additional to correspondence information, for reasoning between data sets. Versions are described for both continuous and categorical labels. Unlike traditional LRA that only calculates an embedding for each data set, this algorithm directly calculates a linear or non-linear map for out-of-sample extensions. To note, this non-linear extension can also be used with other existing manifold learning algorithms, like traditional manifold alignment.

The last contribution of the dissertation is a new machine learning problem domain, spectroscopic data analysis. Within this domain, the task of instrument calibration transfer is introduced. This problem is universal to spectroscopy and of unique importance to space exploration. These data present a new set of challenges to the machine learning community because of their unique characteristics described in this dissertation. The LRA family of algorithms is shown to be effective in this new domain, solving problems in and out of this world.

6.1 Future Work

This problem space has large room for future work, whether adapting existing algorithms for mixtures of manifolds or creating novel mixture-friendly methods. However, for brevity, this section is limited to work directly related to the methods described in this dissertation. Extensions and open problems for the two primary tasks dealt with in this dissertation, alignment and domain adaptation, are offered.

Manifold alignment methods have historically had difficulty scaling to large data sets. Unfortunately, this is true of LRA-based methods as well. When the data sets used are larger than tens of thousands of examples, the block diagonal construction of the matrices causes the eigendecomposition problem, in the algorithm's second step, to become too difficult to solve on a personal computer. This stacking has

been necessary to preserve the intra-set correspondences for reasoning across data sets, while maintaining a closed-form solution. A large number of mid-sized data sets would incur the same difficulty. One possible remedy to the scaling problem could be approximately solving the eigendecomposition (equation 3.13). Approximate spectral decomposition is a well studied problem [49, 56, 75]. One of the most investigated approaches is the Nyström approximation method [22, 34, 50, 92].

Using an approximation technique will likely aid LRA in scaling, but the systemic problem of diagonally stacking data matrices would remain. Fixing this requires a larger design change to the algorithm. Instead of minimizing the loss function in equation 2.2, a new objective would be necessary. For example, the correspondence information could be incorporated into the objective function using a series of soft constraints, which could be solved iteratively.

In the domain adaptation task, it would be advantageous to be able to incorporate importance weights for each data set. In the current description of CADA, when the target data set is significantly smaller than the source sets, the target information may be overwhelmed in the model.

A more ambitious problem in heterogeneous domain adaptation is intra-space transformation. For many real-world tasks, it would be beneficial to be able to transfer samples from one representation to another, using the joint space as an intermediary. This has been described without demonstration in previous manifold alignment literature. For example, to view X_1 in the X_2 representation one would use the mapping $f_1 f_2^{\dagger}$, where $f_2^{\dagger}$ is the pseudo-inverse. This inverse is not well-defined and does not yield a bijection. In practice, this method does not work to transform spectra from one instrument representation to another.

Two potential solutions to this problem are: (1) instead of using a pseudo-inverse of the latent space map, explicitly learn a new smooth map (a second set of weights) from each latent space back to the original feature space, or (2) when calculating the

original latent space mapping function, include a regularizing term that penalizes the smoothness of the inverse maps,

$$\arg\min f^\top x^\top L x f + \frac{1}{2}\left\|x - x f f^\dagger\right\|^2.$$

Deep learning approaches may provide a solution to both the scaling problem and the intra-space transformation problem. In computer vision, a new area of research has developed using deep networks to translate images between representations. For example, given a series of paintings and photographs, or paintings from two stylistically different artists, a deep learning model is used to learn a translation between the representations, with the ability to then generate a painting from a photograph or a painting in another style. Methods have been described that assume the data are in correspondence [45], referred to as paired training, as well as methods that do not assume paired data [52, 98]. Many of these techniques are based upon the generative adversarial network framework [37]. While these techniques were first presented for image data sets, many of their underlying concepts extend to arbitrary data types. However, these methods may fail when the data representations are less comparable. Using these methods to incorporate correspondence information, a deep low rank alignment algorithm could be designed that would provide smooth transformations between data representations. A deep architecture would also alleviate the issue of scaling through process distribution [1, 21].

In addition to fixing some of the shortcomings of LRA, deep learning may provide a better solution to heterogeneous domain adaptation. A non-linear extension of canonical correlation analysis (CCA), called deep CCA (DCCA), has been shown to find better representations than linear CCA or non-linear kernel CCA [4]. Like traditional CCA, DCCA only addressed the two data set. Building upon DCCA and multi-modal deep learning [58], the work was extended to the multi-data set case [46]. Using deep semi-supervised embedding as a base [90], the low rank representation and

correspondence information could be incorporated into a deep architecture. Combining this with a multi-view deep network for cross-view classification [46] could provide a deep solution to heterogeneous domain adaptation.

BIBLIOGRAPHY

[1] Abadi, Martín, Agarwal, Ashish, Barham, Paul, Brevdo, Eugene, Chen, Zhifeng, Citro, Craig, Corrado, Greg S, Davis, Andy, Dean, Jeffrey, Devin, Matthieu, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467* (2016).

[2] Amini, Massih, Usunier, Nicolas, and Goutte, Cyril. Learning from multiple partially observed views-an application to multilingual text categorization. In *Advances in Neural Information Processing Systems* (2009), pp. 28–36.

[3] Ammar, Haitham Bou, Eaton, Eric, Ruvolo, Paul, and Taylor, Matthew E. Unsupervised cross-domain transfer in policy gradient reinforcement learning via manifold alignment. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence* (2015), pp. 2504–2510.

[4] Andrew, Galen, Arora, Raman, Bilmes, Jeff, and Livescu, Karen. Deep canonical correlation analysis. In *International Conference on Machine Learning* (2013), pp. 1247–1255.

[5] Belkin, M., and Niyogi, P. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation 15*, 6 (2003), 1373–1396.

[6] Bengio, Yoshua, and Monperrus, Martin. Non-local manifold tangent learning. In *Advances in Neural Information Processing Systems* (2005), MIT Press, pp. 129–136.

[7] Blei, David M, Ng, Andrew Y, and Jordan, Michael I. Latent dirichlet allocation. *Journal of Machine Learning Research 3*, Jan (2003), 993–1022.

[8] Blitzer, John, Kakade, Sham, and Foster, Dean P. Domain adaptation with coupled subspaces. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics* (2011), pp. 173–181.

[9] Boucher, Thomas, Carey, CJ, Mahadevan, Sridhar, and Dyar, M. Darby. Aligning mixed manifolds. In *Proceedings of the 29th Conference on Artificial Intelligence* (2015), pp. 2511–2517.

[10] Boucher, Thomas, Dyar, M. Darby, and Mahadevan, Sridhar. Proximal methods for calibration transfer. *Journal of Chemometrics 31*, 4 (2017).

[11] Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning 3*, 1 (2011), 1–122.

[12] Candès, E.J., and Tao, T. The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory 56*, 5 (2010), 2053–2080.

[13] Carey, C, Boucher, T, Mahadevan, S, Bartholomew, P, and Dyar, MD. Machine learning tools formineral recognition and classification from raman spectroscopy. *Journal of Raman Spectroscopy 46*, 10 (2015), 894–903.

[14] Caspers, PJ, Lucassen, GW, and Puppels, GJ. Combined in vivo confocal raman spectroscopy and confocal microscopy of human skin. *Biophysical journal 85*, 1 (2003), 572–580.

[15] Chang, Hong, and Yeung, Dit-Yan. Robust locally linear embedding. *Pattern Recognition 39*, 6 (2006), 1053–1065.

[16] Cortes, Corinna, and Mohri, Mehryar. Domain adaptation in regression. In *Proceedings of the 22nd International Conference on Algorithmic Learning Theory* (2011), pp. 308–323.

[17] Cortes, Corinna, and Mohri, Mehryar. Domain adaptation and sample bias correction theory and algorithm for regression. *Theoretical Computer Science 519* (2014), 103–126.

[18] Daszykowski, M, Walczak, B, and Massart, DL. Representative subset selection. *Analytica Chimica Acta 468*, 1 (2002), 91–103.

[19] Daumé, III, Hal, Kumar, Abhishek, and Saha, Avishek. Frustratingly easy semi-supervised domain adaptation. In *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing* (2010), pp. 53–59.

[20] Daumé, III, Hal, and Marcu, Daniel. Domain adaptation for statistical classifiers. *Journal of Artificial Intelligence Research 26*, 1 (May 2006), 101–126.

[21] Dean, Jeffrey, Corrado, Greg, Monga, Rajat, Chen, Kai, Devin, Matthieu, Mao, Mark, Senior, Andrew, Tucker, Paul, Yang, Ke, Le, Quoc V, et al. Large scale distributed deep networks. In *Advances in neural information processing systems* (2012), pp. 1223–1231.

[22] Drineas, Petros, and Mahoney, Michael W. On the Nyström method for approximating a gram matrix for improved kernel-based learning. *The Journal of Machine Learning Research 6*, Dec (2005), 2153–2175.

[23] Duan, Lixin, Xu, Dong, and Tsang, Ivor. Learning with augmented features for heterogeneous domain adaptation. *arXiv preprint arXiv:1206.4660* (2012).

[24] Dyar, M. D., Breitenfeld, L. B., Carey, CJ., Bartholomew, P., Tague, T. J., Wang, P., Mertzman, S., Byrne, S. A., Crowley, M. C., Leight, C., Watts, E., Campbell, J. C., Celestian, A., McKeeby, B., Jaret, S., Glotch, T., Berlanga, G., and Misra, A. K. Interlaboratory and cross-instrument comparison of raman spectra of 96 minerals. In *47th Lunar and Planetary Science Conference* (Houston, 2016), Lunar and Planetary Institute, p. Abstract #2240.

[25] Dyar, M.D., Carmosino, M.L., Breves, E.A., Ozanne, M.V., Clegg, S.M., and Wiens, R.C. Comparison of partial least squares and lasso regression techniques as applied to laser-induced breakdown spectroscopy of geological samples. *Spectrochim. Acta B. 70* (2012), 51–67.

[26] Elhamifar, Ehsan, and Vidal, René. Sparse subspace clustering. In *IEEE Conference on Computer Vision and Pattern Recognition* (2009), pp. 2790–2797.

[27] Favaro, P., Vidal, R., and Ravichandran, A. A closed form solution to robust subspace estimation and clustering. *IEEE Conference on Computer Vision and Pattern Recognition* (2011), 1801–1807.

[28] Fernando, Basura, Habrard, Amaury, Sebban, Marc, and Tuytelaars, Tinne. Unsupervised visual domain adaptation using subspace alignment. In *Proceedings of the IEEE International Conference on Computer Vision* (2013), pp. 2960–2967.

[29] Fernando, Basura, Habrard, Amaury, Sebban, Marc, and Tuytelaars, Tinne. Subspace alignment for domain adaptation. arXiv:1409.5241.

[30] Feudalea, R., Woodya, N., Tana, H., Mylesa, A., Brown, S., and Ferreb, J. Transfer of multivariate calibration models: a review. *Chemometrics and Intelligent Laboratory Systems 64* (2002), 181–192.

[31] Gale, W.A., and Church, K.W. A program for aligning sentences in bilingual corpora. *Computational Linguistics 19* (1993).

[32] Giguere, Stephen, Boucher, Thomas, Carey, CJ, Mahadevan, Sridhar, and Dyar, M Darby. A fully customized baseline removal framework for spectroscopic applications. *Applied spectroscopy 71*, 7 (2017), 1457–1470.

[33] Giguere, Stephen, Carey, CJ, Boucher, Thomas, Mahadevan, Sridhar, and Dyar, M. Darby. An optimization perspective on baseline removal for spectroscopy. *Fifth IJCAI Workshop on AI in Space* (2015).

[34] Gittens, Alex, and Mahoney, Michael W. Revisiting the Nyström method for improved large-scale machine learning. *The Journal of Machine Learning Research 17*, 1 (2016), 3977–4041.

[35] Goldberg, Andrew B., Zhu, Xiaojin, Singh, Aarti, Xu, Zhiting, and Nowak, Robert. Multi-manifold semi-supervised learning. In *Proceedings of the 12th International Conference on Artificial Intelligence and Statistics* (2009), vol. 5, pp. 169–176.

[36] Gong, Dian, Zhao, Xuemei, and Medioni, Grard G. Robust multiple manifold structure learning. In *Proceedings of the 29th International Conference on Machine Learning* (2012).

[37] Goodfellow, Ian, Pouget-Abadie, Jean, Mirza, Mehdi, Xu, Bing, Warde-Farley, David, Ozair, Sherjil, Courville, Aaron, and Bengio, Yoshua. Generative adversarial nets. In *Advances in neural information processing systems* (2014), pp. 2672–2680.

[38] Grauman, Kristen. Geodesic flow kernel for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2012), pp. 2066–2073.

[39] Haidar, Azzam, Luszczek, Piotr, and Dongarra, Jack. New algorithm for computing eigenvectors of the symmetric eigenvalue problem. In *Parallel & Distributed Processing Symposium Workshops (IPDPSW), 2014 IEEE International* (2014), IEEE, pp. 1150–1159.

[40] Halko, N., Martinsson, P. G., and Tropp, J. A. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review 53*, 2 (2011), 217–288.

[41] Ham, J., Lee, D.D., and Saul, L.K. Semisupervised alignment of manifolds. *10 th International Workshop on Artificial Intelligence and Statistics 120-127* (2005).

[42] He, Xiaofei, and Niyogi, Partha. Locality Preserving Projections. *Advances in Neural Information Processing Systems 16* (2004).

[43] Hofmann, Thomas. Probabilistic latent semantic analysis. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence* (1999), Morgan Kaufmann Publishers Inc., pp. 289–296.

[44] Hotelling, Harold. Relations between two sets of variates. *Biometrika 28*, 3-4 (1936), 321–377.

[45] Isola, Phillip, Zhu, Jun-Yan, Zhou, Tinghui, and Efros, Alexei A. Image-to-image translation with conditional adversarial networks. *IEEE Conference on Computer Vision and Pattern Recognition* (2017).

[46] Kan, Meina, Shan, Shiguang, and Chen, Xilin. Multi-view deep network for cross-view classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 4847–4855.

[47] Kennard, Ronald W, and Stone, Larry A. Computer aided design of experiments. *Technometrics 11*, 1 (1969), 137–148.

[48] Koehn, P. Europarl: A parallel corpus for statistical machine translation. *In MT Summit* (2005).

[49] Kumar, Sanjiv, Mohri, Mehryar, and Talwalkar, Ameet. On sampling-based approximate spectral decomposition. In *Proceedings of the 26th annual international conference on machine learning* (2009), ACM, pp. 553–560.

[50] Kumar, Sanjiv, Mohri, Mehryar, and Talwalkar, Ameet. Sampling methods for the nyström method. *Journal of Machine Learning Research 13*, Apr (2012), 981–1006.

[51] Lafon, S., Keller, Y., and Coifman, R. R. Data fusion and multicue data matching by diffusion maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence 28*, 11 (2006), 1784–1797.

[52] Liu, Ming-Yu, Breuel, Thomas, and Kautz, Jan. Unsupervised image-to-image translation networks. In *Advances in Neural Information Processing Systems 30*. 2017, pp. 700–708.

[53] Liu, R., Hao, R., and Su, Z. Mixture of manifolds clustering via low rank embedding. *Journal of Information & Computational Science 8* (2011), 725–737.

[54] Ma, J., Saul, L., Savage, S., and Voelker, G. Identifying suspicious URLs: An application of large-scale online learning. *Proceedings of the International Conference on Machine Learning* (2009), 681–688.

[55] Ma, Y., and Fu, Y. *Manifold Learning Theory and Applications*. CRC Press, 2011.

[56] Mahoney, Michael W, et al. Randomized algorithms for matrices and data. *Foundations and Trends in Machine Learning 3*, 2 (2011), 123–224.

[57] Matijevič, Gal, Prša, Andrej, Orosz, Jerome A, Welsh, William F, Bloemen, Steven, and Barclay, Thomas. Kepler eclipsing binary stars. iii. classification of kepler eclipsing binary light curves with locally linear embedding. *The Astronomical Journal 143*, 5 (2012), 123.

[58] Ngiam, Jiquan, Khosla, Aditya, Kim, Mingyu, Nam, Juhan, Lee, Honglak, and Ng, Andrew Y. Multimodal deep learning. In *Proceedings of the 28th international conference on machine learning (ICML-11)* (2011), pp. 689–696.

[59] Pan, Sinno Jialin, Tsang, Ivor W., Kwok, James T., and Yang, Qiang. Domain adaptation via transfer component analysis. In *Proceedings of the 21st International Jont Conference on Artifical Intelligence* (2009), pp. 1187–1192.

[60] Parikh, Neal, and Boyd, Stephen. Proximal algorithms. *Foundations and Trends in Optimization 1*, 3 (2014), 127–239.

[61] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research 12* (2011), 2825–2830.

[62] Pei, Yuru, Kim, Tae-Kyun, and Zha, Hongbin. Unsupervised random forest manifold alignment for lipreading. In *Proceedings of the IEEE International Conference on Computer Vision* (2013), pp. 129–136.

[63] Resnik, P., and Smith, N.A. The web as a parallel corpus. *Computational Linguistics 29* (2003), 349–380.

[64] Ridder, T., Ver Steeg, B., and Price, G. Robust calibration transfer in noninvasive ethanol measurements, part I: Mathematical basis for spectral distortions in fourier transform near-infrared spectroscopy. *Applied Spectroscopy 68* (2014), 852–864.

[65] Roweis, S., and Saul, L. Nonlinear dimensionality reduction by locally linear embedding. *Science 290* (2000), 2323–2326.

[66] Rull, Fernando, Maurice, Sylvestre, Hutchinson, Ian, Moral, Andoni, Perez, Carlos, Diaz, Carlos, Colombo, Maria, Belenguer, Tomas, Lopez-Reyes, Guillermo, Sansano, Antonio, et al. The raman laser spectrometer for the exomars rover mission to mars. *Astrobiology 17*, 6-7 (2017), 627–654.

[67] Saul, L., and Roweis, S. Think globally, fit locally: Unsupervised learning of low dimensional manifolds. *Journal of Machine Learning Research 4* (2003), 119–155.

[68] Savalle, Pierre-André, Richard, Emile, and Vayatis, Nicolas. Estimation of Simultaneously Sparse and Low Rank Matrices. In *Proceedings of the 29th International Conference on Machine Learning* (2012).

[69] Schneider, Timo, Schauerte, Boris, and Stiefelhagen, Rainer. Manifold alignment for person independent appearance-based gaze estimation. In *22nd International Conference on Pattern Recognition* (2014), IEEE, pp. 1167–1172.

[70] Silveira, Fabricio L, Pacheco, Marcos TT, Bodanese, Benito, Pasqualucci, Carlos A, Zângaro, Renato A, and Silveira, Landulfo. Discrimination of nonmelanoma skin lesions from non-tumor human skin tissues in vivo using raman spectroscopy and multivariate statistics. *Lasers in surgery and medicine 47*, 1 (2015), 6–16.

[71] Soltanolkotabi, Mahdi, Elhamifar, Ehsan, Candes, Emmanuel J, et al. Robust subspace clustering. *The Annals of Statistics 42*, 2 (2014), 669–699.

[72] Souvenir, R., and Pless, R. Manifold clustering. In *10th IEEE International Conference on Computer Vision* (2005), vol. 1, IEEE, pp. 648–653.

[73] Stella, X Yu, and Shi, Jianbo. Multiclass spectral clustering. In *Proceedings of the 9th IEEE International Conference on Computer Vision* (2003), pp. 313–319.

[74] Sun, Baochen, Feng, Jiashi, and Saenko, Kate. Return of frustratingly easy domain adaptation. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence* (2016), pp. 2058–2065.

[75] Talwalkar, Ameet, Kumar, Sanjiv, and Rowley, Henry. Large-scale manifold learning. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on* (2008), IEEE, pp. 1–8.

[76] Tenenbaum, Joshua B., Silva, Vin de, and Langford, John C. A global geometric framework for nonlinear dimensionality reduction. *Science 290*, 5500 (2000), 2319–2323.

[77] Tucker, J.M., Dyar, M.D., Schaefer, M.W., Clegg, S.M., and Wiens, R.C. Optimization of laser-induced breakdown spectroscopy for rapid geochemical analysis. *Chemical Geology 277* (2010), 137–148.

[78] Vanderplas, Jake, and Connolly, Andrew. Reducing the dimensionality of data: Locally linear embedding of sloan galaxy spectra. *The Astronomical Journal 138*, 5 (2009), 1365.

[79] Vidal, Rene, Ma, Yi, and Sastry, Shankar. Generalized principal component analysis (GPCA). *IEEE Transactions on Pattern Analysis and Machine Intelligence 27*, 12 (2005), 1945–1959.

[80] Vinzi, Vincenzo Esposito, Chin, Wynne W., Henseler, Jrg, and Wang, Huiwen. *Handbook of Partial Least Squares: Concepts, Methods and Applications*, 1st ed. Springer Publishing Company, Incorporated, 2010.

[81] Wang, C., and Mahadevan, S. Manifold alignment using procrustes analysis. *Proceedings of the 25th International Conference on Machine Learning* (2008), 1120–1127.

[82] Wang, C., and Mahadevan, S. A general framework for manifold alignment. *AAAI Fall Symposium on Manifold Learning and its App.* (2009).

[83] Wang, Chang, and Mahadevan, Sridhar. Heterogeneous domain adaptation using manifold alignment. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence* (2011), p. 1541.

[84] Wang, Chang, and Mahadevan, Sridhar. Manifold alignment preserving global geometry. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence* (2013), pp. 1743–1749.

[85] Wang, Xuezhi, kuo Huang, Tzu, and Schneider, Jeff. Active transfer learning under model shift. In *Proceedings of the 31st International Conference on Machine Learning* (2014), pp. 1305–1313.

[86] Wang, Xuezhi, and Schneider, Jeff. Flexible transfer learning under support and model shift. In *Advances in Neural Information Processing Systems* (2014), pp. 1898–1906.

[87] Wang, Y., Jiang, Y., Wu, Y., and Zhou, Z. H. Spectral clustering on multiple manifolds. *IEEE Transactions on Neural Networks 22*, 7 (July 2011), 1149–1161.

[88] Wang, Y., Jiang, Y., Wu, Y., and Zhou, Z.H. Multi-manifold clustering. In *PRICAI 2010: Trends in Artificial Intelligence*. Springer, 2010, pp. 280–291.

[89] Watson, G Alistair. Characterization of the subdifferential of some matrix norms. *Linear algebra and its applications 170* (1992), 33–45.

[90] Weston, Jason, Ratle, Frédéric, Mobahi, Hossein, and Collobert, Ronan. Deep learning via semi-supervised embedding. In *Neural Networks: Tricks of the Trade*. Springer, 2012, pp. 639–655.

[91] Wiens, R.C., Maurice, S., Lasue, J., Forni, O., Anderson, R.B., Clegg, S., Bender, S., Blaney, D., Barraclough, B.L., Cousin, A., Deflores, L., Delapp, D., Dyar, M.D., Fabre, C., Gasnault, O., Lanza, N., Mazoyer, J., Melikechi, N., Meslin, P.-Y., Newsom, H., Ollila, A., Perez, R., Tokar, R.L., and Vaniman, D. Pre-flight calibration and initial data processing for the ChemCam laser-induced breakdown spectroscopy instrument on the Mars science laboratory rover. *Spectrochimica Acta Part B: Atomic Spectroscopy 82* (2013), 1–27.

[92] Williams, C.K.I., and Seeger, M. Using the Nyström method to speed up kernel machines. *Advances in Neural Information Processing Systems* (2001).

[93] Workman, Jerome J. A review of calibration transfer practices and instrument differences in spectroscopy. *Applied spectroscopy 72*, 3 (2018), 340–365.

[94] Yamada, Makoto, Sigal, Leonid, and Chang, Yi. Domain adaptation for structured regression. *International Journal of Computer Vision 109*, 1-2 (2014), 126–145.

[95] Yang, Qiang, Pan, Sinno Jialin, and Zheng, Vincent Wenchen. Estimating location using Wi-Fi. *IEEE Intelligent Systems 23*, 1 (2008), 8–13.

[96] Yu, Yao-Liang. On decomposing the proximal map. In *Advances in Neural Information Processing Systems* (2013), pp. 91–99.

[97] Zhou, Joey Tianyi, Tsang, Ivor W, Pan, Sinno Jialin, and Tan, Mingkui. Heterogeneous domain adaptation for multiple classes. In *Artificial Intelligence and Statistics* (2014), pp. 1095–1103.

[98] Zhu, Jun-Yan, Park, Taesung, Isola, Phillip, and Efros, Alexei A. Unpaired image-to-image translation using cycle-consistent adversarial networks. *IEEE International Conference on Computer Vision (ICCV)* (2017).