DISCRIMINATIVE MANIFOLD EMBEDDING WITH IMPRECISE, UNCERTAIN, AND AMBIGUOUS DATA

By

CONNOR H. MCCURLEY

A ORAL QUALIFYING EXAM PROPOSAL PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

2020

TABLE OF CONTENTS

LIST OF TABLES

Table page

LIST OF FIGURES

LIST OF ABBREVIATIONS

CT	Computed Tomography
CHL	Competitive Hebbian Learning
DR	Dimensionality Reduction
DSIAC	Defense Systems Information Analysis Center
EM	Expectation-Maximization
FA	Factor Analysis
FP	False Positive
FPS	Frames per Second
GLVM	General Latent Variable Model
GTM	Generative Topographic Mapping
GPS	Global Positioning System
HS	Hyperspectral
HSI	Hyperspectral Image
Isomap	Isometric Feature Mapping
LDA	Fisher's Linear Discriminant Analysis
LE	Laplacian Eigenmaps
LiDAR	Light Detection and Ranging
LFW	Labeled Faces in the Wild Dataset
LLE	Locally Linear Embedding
LMNN	Large-Margin K-Nearest Neighbors
LPP	Locality Preserving Projections
MDS	Multi-dimensional Scaling
MIL	Multiple Instance Learning
MIDR	Multiple Instance Dimensionality Reduction
MWIR	Mid-wave Infrared
NCA	Neighborhood Component Analysis
PCA	Principal Component Analysis
RBF	Radial Basis Function
ROI	Region of Interest
S-LE	Supervised Laplacian Eigenmaps
SOA	State-of-the-Art
SOM	Self-organizing Feature Map
SVD	Singular Value Decomposition
TP	True Positive
t-SNE	t-Distributed Stochastic Neighbor Embedding
UMAP	Uniform Manifold Approximation and Projection

LIST OF SYMBOLS AND NOMENCLATURE

$\mathcal{E}$	Set of edges of an undirected, weighted graph
$\mathcal{G}$	Undirected, weighted graph
$\mathcal{V}$	Set of vertices in an undirected, weighted graph
$\boldsymbol{W}$	Adjacency of affinity matrix
$\boldsymbol{X}$	Data matrix
$\mathcal{X}$	Riemannian manifold

CHAPTER 1
INTRODUCTION

Target detection is a paramount area of research in the field of remote sensing which aims to locate an object or region of interest while suppressing unrelated objects and information. (Geng et al., 2017; Chaudhuri and Parui, 1995). Target detection can be formulated as a two-class classification problem where samples belonging to a class of interest are discriminated from a background distribution (Zare et al., 2018). The goal of target detection in remote sensing is to correctly classify every true positive instance (TP) in a given scene while indicating no false alarms (FA) (non-target samples predicted as targets). However, there is always a trade-off in performance, and one might actually choose to miss targets to achieve a lower false-alarm rate (Weinberger), and vice versa. Many remote sensing target detection techniques in the literature are variations of constrained energy minimization or maximum likelihood with matched filters (Geng et al., 2017; Chaudhuri and Parui, 1995). The goal of learning in these scenarios is to discover prototypes which represent both the target and background classes that can be used to classify a sample at test. Traditional supervised learning approaches such as these require extensive amounts of highly precise, sample- or pixel-level groundtruth to guide algorithmic training. However, acquiring large quantities of accurately labeled training data can be expensive both in terms of time and resources, and in some cases, may even be infeasible to obtain. To demonstrate these inherent labeling problems, consider the following real-world remote sensing examples, beginning with the hyperspectral target detection scenario described in (Du, 2017) and (Bocinsky, 2019):

Hyperspectral (HS) sensors collect spatial and spectral information of a scene by receiving radiance data in hundreds of contiguous wavelengths (Zare, 2008). Due to their inherent properties, HS cameras can provide a broad range of spectral information about the materials present in a scene, and are thus useful for detecting targets whose spectral signatures vary from the background. Such examples include airplanes on a tarmac or weeds in a cornfield.

While HS provides nice properties for target detection, there are significant challenges when utilizing this modality. First, the spatial resolution of HS cameras can be low. As an example, some HS cameras have spatial resolution of $30m^2$ when capturing scenes from the air. This implies that objects of interest in a scene, such as an airplane, will actually be contained in a single pixel along with other materials, such as asphalt. When performing target detection/recognition on that pixel, the HS spectra will not be distinguishable as a single, pure material, but as a sub-pixel mixture where the actual materials present as well as their corresponding proportions are unknown. Second, assuming pure target pixels are available, accurate positioning at the desired resolution may not be. For example, when analyzing a scene from an airplane or satellite, it is necessary to denote the true locations of targets on the ground using a global positioning system (GPS). It is not uncommon, however, to experience GPS error of greater magnitude than the HS pixel-level spatial resolution. This implies that a halo of uncertainty potentially surrounds every target pixel in the hyperspectral image (HSI), thus making labeling on the pixel-level difficult.

This example demonstrates inherent infeasibility to obtain accurate sample-level labels due to sensor restrictions on both resolution and accuracy. Furthermore, label imprecision and ambiguity can often be presented from subjectivity between annotators. Many applications such as medical diagnosis and wildlife identification require domain experts to provide accurate data labels. However, there might not always be agreement between expert annotators and humans are prone to making mistakes. For example, when looking at computed tomography (CT) scans for malignant/benign tumors, many doctors would likely determine different pixel-level boundaries denoting a tumor, and in some cases, might even misclassify the detriment of the growth. Similarly, expert wildlife ecologists determining the identity of birds solely from their songs might be uncertain of a species due to corruptive background noise in the audio segment.

Finally, consider the scenarios shown in Figures 1-1 and 1-2. These figures show frames taken from the DSIAC MS-003-DB dataset (DSIAC, 2014) which demonstrates mid-wave

infrared (MWIR) video segments of moving military vehicles taken at approximately 30 frames per second (FPS). Many computer vision algorithms have already been developed to perform target detection using canonical bounding boxes (shown in green in Figure 1-1) (Redmon and Farhadi, 2018). However, drawing tight boxes around targets in each video frame is extremely tedious and time consuming. It would be beneficial if an annotator could provide a less-restrictive form of label, such as a relaxed bounding box (shown in blue in Figure 1-1 and bottom left in Figure 1-2) or as a small subset of target pixels such as single dot or scribble as shown in Figure 1-2. Labeling burden could be reduced even further if a single frame could be labeled at a high level as "including" or "excluding" target pixels, as shown in Figure 1-3.

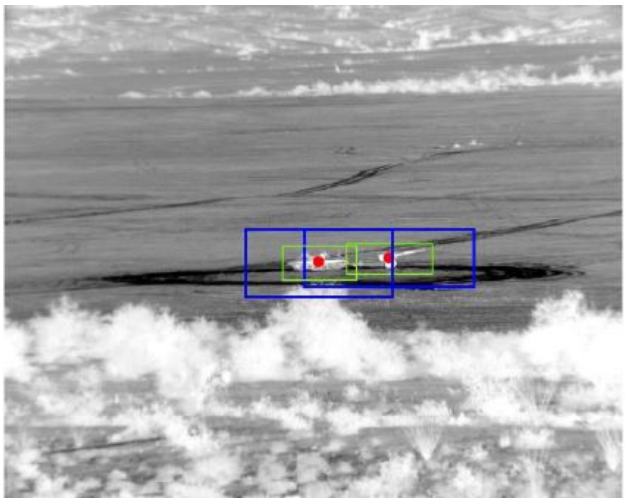


Figure 1-1: A sample frame from the DSIAC MS-003-DB MWIR dataset. Two targets are shown with canonical bounding boxes (green) and relaxed bounding boxes (blue). Red dots represent the centers of the target objects.

Techniques which can address these forms of label ambiguity while achieving comparable or better target detection than standard supervised methods can greatly ease the burdens associated with many remote sensing labeling tasks and allow for the application of pattern recognition techniques which would otherwise be infeasible.

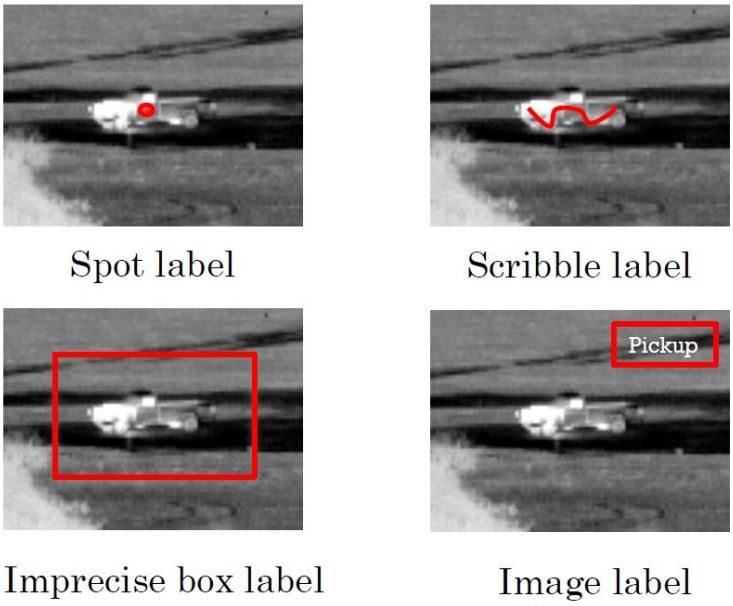


Figure 1-2: Examples of weakly-labeled infrared imagery. The images demonstrate various forms of weak groundtruth around a pickup truck taken with a mid-wave infrared camera. The images show spot, scribble, imprecise bounding box and image-level labels, respectively.

(a) (b)

Figure 1-3: Example of image-level labels for binary target detection. Image (a) is denoted to contain pixels belonging to the target class somewhere within the image, while image (b) clearly contains samples solely from the background distribution.

Learning from uncertain, imprecise and ambiguous data has been an active area of research since the late 1990s and is known as *multiple instance learning* (MIL) or *weak learning* (Bocinsky, 2019). Supervised learning assumes that each training sample is paired with a corresponding classification label. In multiple instance learning, however, the label of each sample is not necessarily known. Instead, MIL approaches learn from groups of data points called *bags*, and each bag concept is paired with a label (Cook, 2015). Under the two-class classification scenario the bags are labeled as *negative* if all data points (or *instances*) are

known to belong to the background class (not the class of interest). While the actual number of positive and negative instances may be unknown, bags are labeled *positive* if *at least one* instance is known to belong to the target class (also called a "true positive") (Zare et al., 2018) or to contain a proportion of true target. The goal of learning under the MIL framework is to train a model which can classify a bag as positive or negative (bag-level classification) or to predict the class labels of individual instances (instance-level classification). Consider again the example shown in Figure 1-3. The figure labeled as "Target" could be considered as a positive bag because, while the image is not accompanied with pixel-level labels, it is known that a true target pixel exists somewhere within the set. Additionally, the image denoted as "Background" could be considered as a negative bag, since it does not contain any pure target pixels. Given data of this type, multiple instance learning could be used to discover target and/or background class representatives which could be used for class discrimination, or a classifier could be trained to label and unseen test image as containing or excluding target pixels (bag-level classification) or to label each individual pixel as belonging to the target class (instance-level classification). As with this example, the MIL problem formulation fits many remote sensing scenarios and is thus an important area of investigation (Du, 2017).

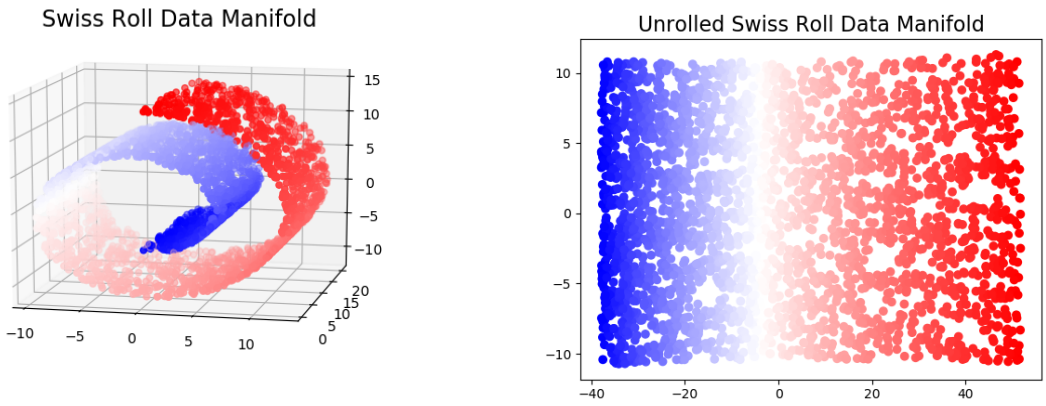
Multiple instance learning approaches in the literature can be broadly generalized into two categories: learning a concept(s) which effectively describes the positive and/or negative classes, or embedding bags in high-dimensional spaces such that positive and negative bags are well-separated. In both cases a classifier can be trained to discriminate bags or individual instances. **Existing approaches in the literature have two primary limitations: (1) MIL methods assume that target instances are already separable in the feature space or that a signature(s) can be learned which represents the target class well, while poorly representing the entirety of the negative class, regardless of potential distribution overlap (Carbonneau et al., 2016). (2) Current methods are unable to address learning complications associated with high-dimensionality.**

These two detrimental limitations can potentially be resolved with the application of *manifold learning*, also commonly referred to as *dimensionality reduction* (DR), *feature embedding*, *geometric machine learning* or *representation learning* in the literature. The goal of manifold learning can be posed as discovering intrinsic (often lower-dimensional) features from the data which meet an overarching objective, such as: preserving variance, finding compressed representations of data, maintaining global or local structure or promoting discriminability in the embedded space (van der Maaten et al., 2007; Bengio et al., 2012; X. Geng et al., 2005; Thorstensen, 2009). Manifold learning and dimensionality reduction have been studied extensively in the literature and have been used for visualization, classification, redundancy removal, compression and data management, improving computational tractability and efficiency, combating the curse of dimensionality and obtaining more appropriate measures of dissimilarity (Bishop et al., 1998; Nickel and Kiela, 2017; Talmon et al., 2015; Tenenbaum et al., 2000; X. Geng et al., 2005; Palomo and Lopez-Rubio, 2017; Kohonen, 1990; Kegl et al., 2008; Bengio et al., 2012).

Dimensionality plays a significant role in determining class separability. Enough features should be incorporated as to adequately describe a class of interest, yet too many features may introduce redundancy, and thus be detrimental to the learning process. The curse of dimensionality states that the number of samples needed to characterize the space spanned by an entity grows exponentially with dimensionality (Murphy, 2012; Theodoridis and Koutroumbas, 2008). This fact is overwhelming in the context of remote sensing, as it is often both difficult to acquire large quantities of labeled data and there are often only few samples available to describe the target class (such as in air- or space-born HSI target detection). Additionally, dissimilarity metrics often break down in high dimensional spaces, making application of traditional classifiers difficult. *Therefore, it is intuitive that approaches should be developed which can combat the curse of dimensionality while also addressing the problems associated with uncertain, imprecise, and ambiguously labeled training data.*

The underlying assumption in using manifold learning for discrimination is that opposing classes either reside on separate manifolds or different regions of a joint, intrinsic manifold, where samples of the same class are close and samples from disparate classes are metrically far. Essentially, the governing class distributions can be described by hyper-surfaces which follow constraints on properties such as continuity and smoothness (Belkin and Niyogi, 2004). The goal of learning in this scenario is to discover embedding functions from the input feature space to a lower-dimensional embedding space where the transformed feature representations allow for improved class discrimination.

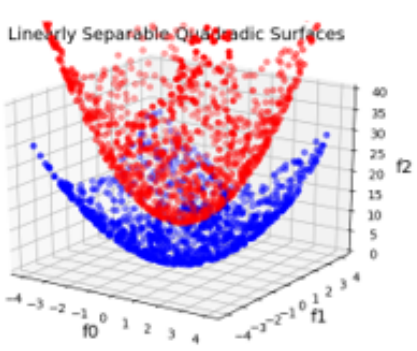
Methods for representation learning have recently gained in popularity because they typically result in high levels of classification accuracy. Some of these methods learn features in a supervised manner to obtain more discriminative representations. However, this learning cannot be done directly in MIL because of the uncertainty on the labels. Thus, *adaptation of discriminative feature learning methods would be beneficial to MIL* (Carbonneau et al., 2016), yet this area of research has scarcely been explored. The first true experimentation was performed in (Sun et al., 2010). In this work, Sun et al. showed that Principal Component Analysis (PCA) failed to incorporate bag-level label information and thus provided poor separation between positive and negative bags. Additionally, Linear Discriminant Analysis (LDA) was used to project bags into a latent space which maximized between-bag separation, while minimizing within-bag dissimilarity. However, LDA often mixed the latent bag representations due to the uncertainty of negative sample distributions in the positive bags. To address these issues, Sun proposed Multiple Instance Dimensionality Reduction (MIDR) which optimized an objective through gradient descent to discover sparse, orthogonal projection vectors in the latent space. Their approach relied on fitting a distribution of negative instances and applying maximum likelihood estimation. This approach was later extended in (Zhu et al., 2018) in attempt to improve sparsity. Both of these methods rely on fitting a distribution of negative instances and determining positive instances as samples which have low likelihood of belonging to the distribution. However, these approaches fail when the the target and background instances are



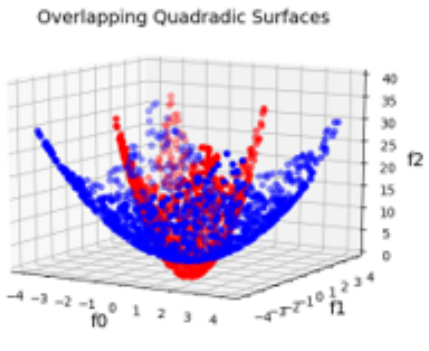
(a) (b)

Figure 1-4: (a) This dataset is known as the Swiss Roll and it depicts a 2-dimensional manifold embedded in 3 dimensions. (b) The Swiss Roll unfolded according to the geodesic path around the manifold.

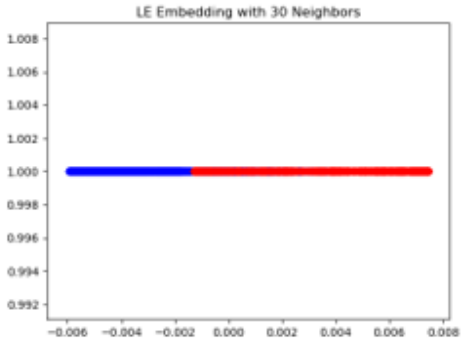
metrically similar in the feature space. For example, consider the data shown in Figure 1-4. This image demonstrates the popular Swiss Roll dataset, which is a 2-dimensional manifold folded in 3-dimensions. Assume that the data classes lie on separate ends of the manifold, and that samples are governed by a smooth labeling function. In Figure 1-4a, the red samples denote one class while the blue represent another. If using Euclidean (straight line) distance to measure dissimilarity, it would appear that many red samples are (untruly) close to the blue. However, if an alternative distance metric such as geodesic distance (around the curve) is used to measure dissimilarity between samples, the true class distributions are better represented. Figure 1-4b shows the unfolding of manifold using geodesic distance. It can be observed that, after the unfolding, the classification problem was transformed from a nonlinear to a linear one. Additionally, a dimension of the data was also deemed unnecessary. It would, therefore, be beneficial to first transform the instance representations into more discriminable forms. Most existing approaches in the literature extend LDA to distinguish between positive and negative bags (Chai et al., 2014; Zhu et al., 2018). These methods typically rely on costly optimization procedures to maximize an objective for orthogonal and sparse projection vectors. The work in (Xu et al.) investigated metric learning on sets of data (where each bag was a set) to learn



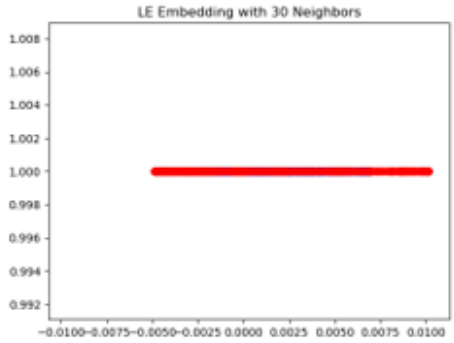
(a)



(b)



(c)

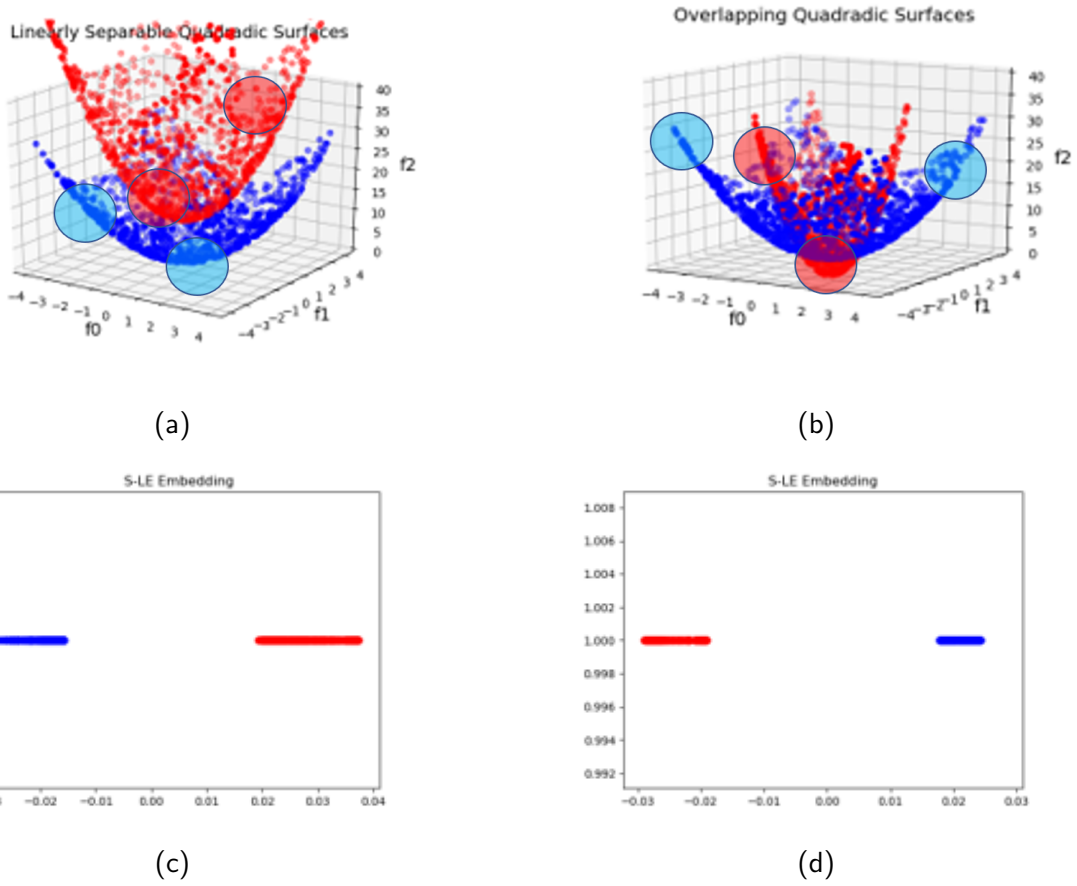


(d)

Figure 1-5

an appropriate similarity metric to compare bags. They do not go as far as to discriminate positive instances within the positive bags, nor do they propose positive target concepts. In fact, virtually all MIL DR methods in the literature are applied, solely, to predict bag-level labels. While bag-level label prediction is useful in many remote sensing applications, such as region of interest (ROI) proposal for anomaly detection, they limit the the ability of learners to classify on the pixel or sample level. Xu et al. proposed an importance term to weight samples believed to be true target exemplars, as those instances are more important in determining the bag-level label (Xu et al.). However, to the author's knowledge, no work has been done to investigate instance-level discrimination through manifold learning/ dimensionality reduction.

To address these points, I propose the following. During this project, techniques will be explored for use in instance-level classification given uncertain and imprecise groundtruth.



(a) (b)

(c) (d)

Figure 1-6

These methods will be developed as universal approaches for discriminative manifold/feature representation learning and dimensionality reduction and will be evaluated on a variety of sensor modalities, including: mid-wave IR, visible, hyperspectral and multispectral imagery, LiDAR and more. *The aim of this project is to develop dimensionality reduction methods which promote class discriminability and are simultaneously capable of addressing uncertainty and imprecision in training data groundtruth.* The motivating idea is to facilitate instance/concept proposition by increasing instance discriminability under the constraints of multiple instance learning. Roughly, the following research questions will be addressed during the scope of this project:

1. Supervised and semi-supervised manifold learning have proven effective at discovering low-dimensional data representations which provide adequate class separation in the

latent space. However, only a handful of manifold learning procedures consider data that is weakly or ambiguously labeled. To address this gap in the literature, a method for weakly-supervised manifold learning will be developed. How does this method of manifold construction compare to state-of-the-art (SOA) manifold learning techniques as well as alternative MI dimensionality reduction methodologies for instance-level label prediction?

2. In conjunction with dimensionality, the use of metric embedding has been shown to promote class separability in the latent space. However, metric embedding typically requires knowledge of instance-level labels. Using only weak, bag-level labels, a method for metric embedding will be developed and utilized with the manifold learning approach in Objective 1 to potentially improve class separation of individual instances. Additionally, a procedure to select the most influential examples for training will be developed.

3. Do the proposed methods facilitate concept learning/selection? Using alternative state-of-the-art MIL approaches, are the selected target instances/concepts more discriminable with the proposed methods than without? How do the proposed methods compare to the alternatives in terms of representation dimensionality, computational complexity, and promotion of discriminability?

Experiments will be conducted on both synthetic data and real applications such as target detection, scene understanding and semantic segmentation in remote sensing imagery. Datasets will include the DSIAC MS-003-DB Algorithm Development Database, MUUFL Gulfport, Faces in the Wild (LFW) and benchmark MIL classification datasets (DSIAC, 2014; Gader et al., 2013; Du and Zare, 2017; Glenn et al.; Huang et al., 2007). Initial results demonstrate the aptitude of the proposed approaches and suggest further development and evaluation of these methods.

CHAPTER 2
BACKGROUND

This chapter provides a literature review of the Multiple Instance Learning framework for learning from weak and ambiguous annotations. A review is provided on Manifold Learning, including classic approaches, supervised and semi-supervised methods and uses of manifolds for functional regularization. Additionally, this chapter reviews the existing literature on metric embedding, focusing heavily on the utilization of contrastive and triplet-based loss evaluation. Reviews describe basic terminology and definitions. Foundational approaches are elaborated and advances are addressed.

2.1 Multiple Instance Learning

Multiple Instance Learning (MIL) was originally proposed in (Dietterich et al., 1997) as a method to handle inherent observation difficulties associated with drug activity prediction. This problem, among many others, fits well into the framework of MIL where training labels are associated with sets of data points, called *bags* instead of each individual data points, or *instances*. Under the *standard MIL assumption*, a bag is given a "positive" label if it is known that *at least one* sample in the set represents pure or partial target. Alternatively, a bag is labeled as "negative" if does not contain any positive instances (Carbonneau et al., 2016). Let $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N] \in \mathbb{R}^{D \times N}$ be training data where D is the dimensionality of an instance, $\boldsymbol{x}_n$, and N is the total number of training instances. The data is grouped into K *bags*, $\boldsymbol{B} = \{\boldsymbol{B}_1, \ldots, \boldsymbol{B}_K\}$, with associated binary bag-level labels, $\mathcal{L} = \{L_1, \ldots, L_K\}$ where

$$L_k = \begin{cases} 1, & \exists \boldsymbol{x}_{kn} \in \boldsymbol{B}_k^+ \ni l_{kn} = 1 \\ -1, & l_{kn} = 0 \quad \forall \boldsymbol{x}_{kn} \in \boldsymbol{B}_k^- \end{cases} \tag{2-1}$$

and $\boldsymbol{x}_{kn}$ denotes the n^{th} instance in positive bag $\boldsymbol{B}_k^+$ or negative bag $\boldsymbol{B}_k^-$ (Zare et al., 2018) and l_{kn} denotes the instance-level label on instance $\boldsymbol{x}_{kn}$. Figure 2-1 demonstrates the concept of MIL bags. The objective of learning under MIL is, given only bag-level label information, to

fit a model which can classify bags as either being positive or negative (bag-level classification) or to predict the class labels of individual instances (instance-level classification).

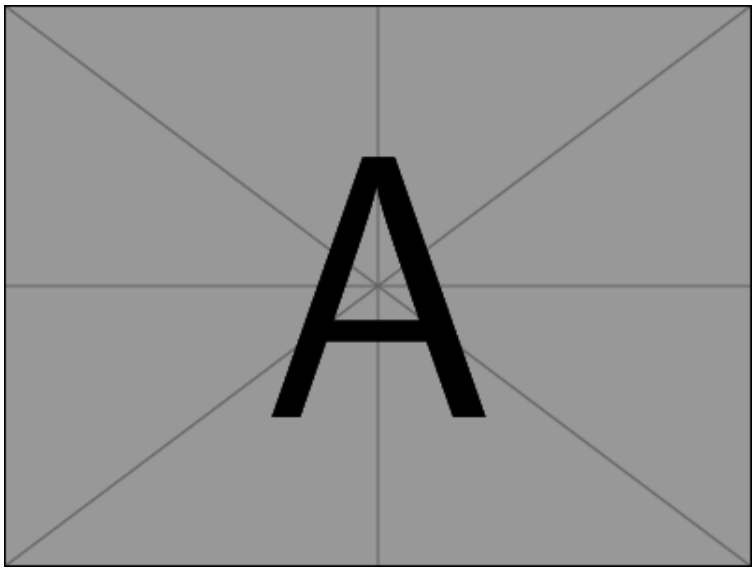


Figure 2-1: Placeholder for examples of positive and negative bag concepts

2.1.1 Multiple Instance Space Paradigm

Multiple Instance Learning via Embedded Instance Selection

2.1.2 Multiple Instance Concept Learning

2.1.3 Multiple Instance Classification

2.1.4 Multiple Instance Boosting

2.2 Manifold Learning

Real-world remote sensing data such as hyperspectral imagery, ground-penetrating radar scans and sonar signals are naturally represented by high-dimensional feature vectors. However, in order to handle such real-world data adequately, its dimensionality usually needs to be reduced (van der Maaten et al., 2007; Belkin and Niyogi, 2004). The problem considered in this work is discovering feature representations that promote class discriminability for target or anomaly detection. This is typically achieved in one of two ways. First, features can be projected into a high-dimensional space (such as a Kernel Hilbert Space) using a kernel

function. The second option, which is the focus of this work, is to transform the data into a new (often lower-dimensional) coordinate system which optimizes feature representations for discrimination (Vural and Guillemot, 2018).

The application of *dimensionality reduction* (DR) has proven useful in myriad applications in the literature, such as: visualization of high-dimensional data, classification, redundancy removal, compression and data management, improving computational tractability and efficiency, and reducing the effects of the Curse of Dimensionality (Bishop et al., 1998; Nickel and Kiela, 2017; Talmon et al., 2015; Tenenbaum et al., 2000; X. Geng et al., 2005; Palomo and Lopez-Rubio, 2017; Kohonen, 1990; Kegl et al., 2008; Bengio et al., 2012). In classification of object entities, it is often assumed that classes can be described by an *intrinsic* subset of representative features which demonstrate geometrical structure (Belkin et al., 2006). These structures are called intrinsic *manifolds*, and they represent the generating distributions of class objects exactly by the number of degrees of freedom in a dataset (Thorstensen, 2009; Belkin and Niyogi, 2004). Consider the example shown in Figure 2-2. This classic example demonstrated in (Thorstensen, 2009) shows samples from a pose-estimation dataset **CITE**. While each individual image is represented by a vector of features (pixel intensities in this case) in $\mathbb{R}^{4096}$, the dataset only exhibits three degrees of freedom: 1 light variation parameter and 2 rotation angles. Thus, it is intuitive that the dataset lies on a smooth, intrinsic submanifold spanning three dimensions which inherently capture the degrees of freedom in the data.

The goal of manifold Learning is then to discover embedding functions which take data from the input feature space and transform it into a lower-dimensional (ideally intrinsic) coordinate system (also called a *latent space* in the literature) which captures the "useful" properties of the data, while enforcing constraints such as smoothness (the transformation function should not produce sporadic images), continuity (no discontinuous points on the hyper-surface), topological ordering (neighbors in the input space should also be neighbors in the embedded space) or class separability (samples from the same class should fall metrically

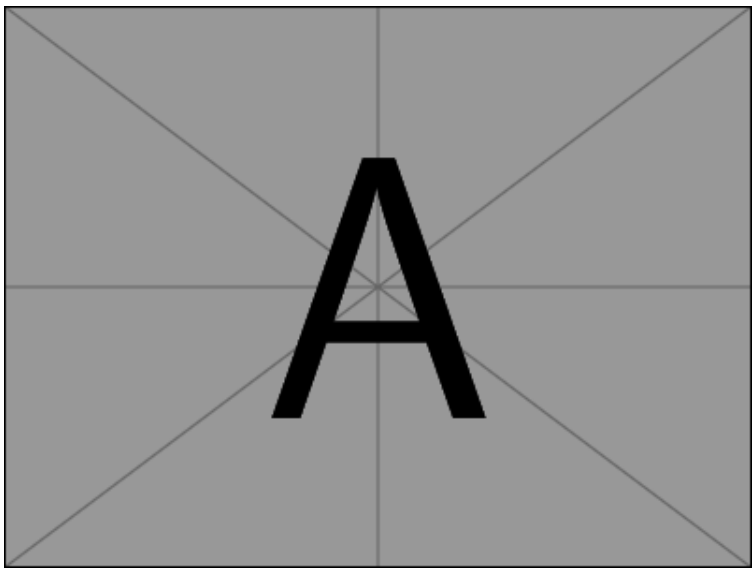


Figure 2-2: Placeholder for example of high D data lying on a low-dimensional sub-manifold.

close to each other in the embedded space and disparate classes should be distinctly far) (Vural and Guillemot, 2018).

This dissertation focuses on investigating the use of manifold learning to increase instance discriminability in the latent space, where labels are solely provided at the bag-level. While there is an expansive literature in unsupervised manifold learning methods, this document will pay special attention to both strictly- and semi-supervised methods, since they are typically adaptations of unsupervised approaches, as well as manifold learning under the MIL framework.

2.2.1 Definition and General Notation

Most studies perform classification or regression after applying unsupervised dimensionality reduction. However, it has been shown that there are advantages of learning the low-dimensional representations and classification/regression models simultaneously (Chao et al., 2019; Rish et al., 2008). Considering classification as the main goal of dimensionality reduction, this section provides a summary of the current literature in the area.

Given a data matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N] \in \mathbb{R}^{D \times N}$ where N is the total number of samples and D is the dimensionality of the input feature space, general dimensionality reduction seeks to find a representation $\boldsymbol{Z} \in \mathbb{R}^{d \times N}$ with $d \ll D$ that enhances the between-class separation

while preserving the intrinsic geometric structure of the data(Vural and Guillemot, 2018). In other words, it is assumed that the data lie on a smooth manifold $\mathcal{X}$, which is the image of some parameter domain $\mathcal{Z} \subset \mathbb{R}^d$ under a smooth mapping $\Psi : \mathcal{Z} \to \mathbb{R}^D$. The goal of manifold learning is to discover an inverse mapping to the low-dimensional pre-image coordinates $\boldsymbol{z}_n \in \boldsymbol{z}$ corresponding to points $\boldsymbol{x}_n \in \boldsymbol{X}$. The data matrices $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N]$ and $\boldsymbol{Z} = [\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N]$ are of size $D \times N$ and $d \times N$, respectively. Since these low-dimensional data representations are unknown, they are often referred to as *latent* vectors and the span in $\mathbb{R}^d$ is sometimes called the *latent feature space* or *latent space* for conciseness **CITE**. The primary difference between traditional, unsupervised manifold learning and supervised approaches is that, in supervised manifold learning, data matrix $\boldsymbol{X}$ is accompanied with a corresponding label vector $\boldsymbol{l} = [l_1, \ldots, l_N]$ indicating the corresponding class labels of each sample in $\boldsymbol{X}$.

Manifold learning methods can be subdivided into a wide taxonomy of approaches, with *linear* and *nonlinear* at the root. Nonlinear approaches can be further divided into purely global methods and approaches that capture global structure solely from local information. We begin with a review of popular linear manifold learning techniques before moving into the realm of nonlinear approaches. Base, unsupervised methods are reviewed along with corresponding supervised and semi-supervised adaptations.

2.2.2 Comparison Table of Manifold Learning Methods

2.2.3 Geneology Image of Manifold Learning Methods from van der Maaten 2009, page 2

2.2.4 Linear Manifold Learning

A review of linear manifold learning approaches is provided. Linear approaches are advantageous over nonlinear because they allow for out-of-sample extensions. In other words, linear transformation matrices are learned which can be easily applied on data not included in the training set. However, linear approaches are limited in their abilities to capture irregular data surfaces **CITE**. Principal Component Analysis (PCA), Multi-dimensional Scaling (MDS) and Fisher's Linear Discriminant Analysis (LDA) are reviewed. General approaches are

discussed and supervised as well as nonlinear extensions are elaborated. Special focus is given to (LDA), as it is the only inherently-supervised technique out of the included approaches.

2.2.4.1 Principal Component Analysis (PCA)

Unsupervised PCA. Principal Component Analysis (PCA) is arguably the most popular (and best-studied) technique for dimensionality reduction and manifold learning. It attempts to learn an orthogonal projection of the input data into a lower-dimensional space, known as the principal subspace, such that the variance of the projected data is maximized (Chao et al., 2019). In other words, each *principal axis*, or *principal component*, of the learned coordinate system is orthogonal to the other principal components. In summary, the problem of PCA is to discover basis vectors which linearly combine to reconstruct the data. In practice, data in the input feature space are projected into a new coordinate system of d dimensions, such that the variance along each principal axis is maximized and the reconstruction errors of the data are minimized in the mean-square sense (Thorstensen, 2009). Let V be a d-dimensional subspace of $\mathbb{R}^D$ and let $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_D$ be an orthonormal basis of $\mathbb{R}^D$ such that $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_d$ is a basis of V. The goal of PCA is to find an orthogonal set of basis vectors $\boldsymbol{w}_n \in \mathbb{R}^D$ and corresponding latent coordinates $\boldsymbol{x}_n \in \mathbb{R}^d$ such that the average reconstruction error is minimized (Murphy, 2012)

$$J(\boldsymbol{W}, \boldsymbol{Z}) = \frac{1}{N} \sum_{n=1}^{N} ||\boldsymbol{x}_n - \hat{\boldsymbol{x}}_n||^2 \tag{2-2}$$

where $\hat{\boldsymbol{x}}_n = \boldsymbol{W}\boldsymbol{z}_n$, subject to the constraint that $\boldsymbol{W}$ is *orthonormal*, or that $\boldsymbol{w}_i^T \boldsymbol{w}_j = 0, \forall i \neq j$ and $\boldsymbol{w}_i^T \boldsymbol{w}_i = 1$. This is equivalently written as

$$J(\boldsymbol{W}, \boldsymbol{Z}) = ||\boldsymbol{X} - \boldsymbol{W}\boldsymbol{Z}||_F^2 \tag{2-3}$$

where $\boldsymbol{Z}$ is a $N \times d$ matrix with the $\boldsymbol{z}_n$ in its rows and $||\boldsymbol{A}||_F$ is the *Frobenius norm* of matrix $\boldsymbol{A}$, defined by

$$||\boldsymbol{A}||_F = \sqrt{\sum_{m=1}^{M}\sum_{n=1}^{N} a_{mn}^2} = \sqrt{tr(\boldsymbol{A}^T\boldsymbol{A})} = ||\boldsymbol{A}(:)||_2 \tag{2-4}$$

As noted by Murphy (Murphy, 2012), the optimal solution is obtained by setting $\hat{\boldsymbol{W}} = \boldsymbol{U}_d$, where $\boldsymbol{U}_d$ contains the eigenvectors corresponding to the d largest eigenvalues of the mean-subtracted, empirical data covariance matrix, $\hat{\boldsymbol{S}} = \frac{1}{N}\sum_{n=1}^{N}(\boldsymbol{x}_n - \hat{\boldsymbol{\mu}})(\boldsymbol{x}_n - \hat{\boldsymbol{\mu}})^T$, where $\hat{\boldsymbol{\mu}}$ is the empirical data mean. Therefore, the low-dimensional encoding of the data is given by $\boldsymbol{z}_n = \hat{\boldsymbol{W}}^T\boldsymbol{x}_n$, which is the orthogonal, linear projection of the data onto the column space spanned by the eigenvectors of the d largest eigenvalues of the empirical data covariance. Alternatively,

Assumes Gaussian distributions Typically, the data is standardized before applying PCA, as it can be misled by directions in which the variance is high simply because of the measurement scale.

The example shown in figure **REFERENCE FIGURE** demonstrates the projection of 2-dimensional data onto the first principal axis. As can be seen from the figure, the first principal axis corresponds to the direction of maximal variance of the data. PCA from the viewpoint of variance maximization is often called the *analysis view* of PCA (Murphy, 2012).

PCA has been successfully applied to a large number of domains such as face recognition, coin classification and seismic series analysis (van der Maaten et al., 2007). However PCA suffers from a few drawbacks. First, the dimensionality of the covariance matrix is proportional to the dimensionality of the data points. As a result, the computation of the eigenvectors may be infeasible or untrustworthy (singularity) for high-dimensional data. Additionally, PCA focuses mainly on preserving large pairwise distances between data samples instead of retaining local relationships, which may be important in certain applications **SHOW SWISS ROLL EXAMPLE AGAIN?**.

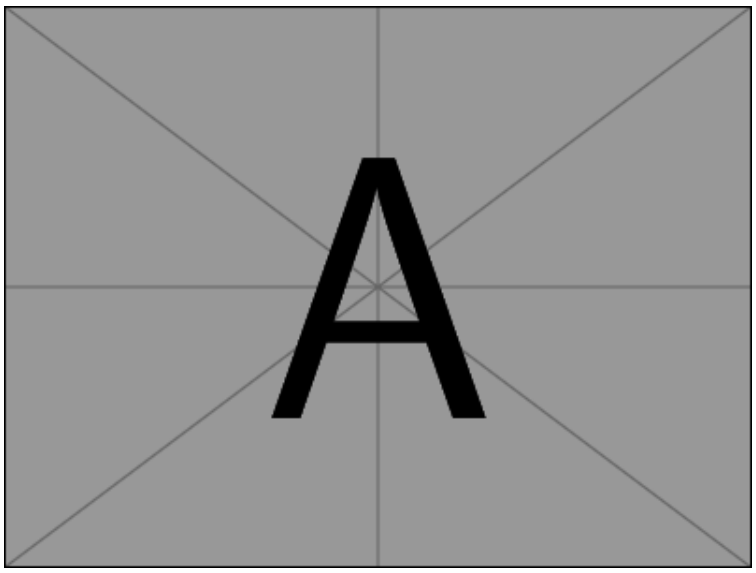


Figure 2-3: Placeholder for PCA projection example.

Many extensions and alternative viewpoints have been made to PCA, such as creating a nonlinear version, a supervised version and looking at it as a factor analysis problem **CITE**. A few adaptations to PCA are discussed in later sections.

Kernel PCA.

Page 497 Murphy textbook

2.2.4.2 Multi-Dimensional Scaling (MDS)

Unsupervised MDS.

Supervised MDS.

Kernel MDS.

2.2.4.3 Fisher's Linear Discriminant Analysis (LDA)

Linear LDA.

Kernel LDA (KDA).

2.2.4.4 Locality Preserving Projection (LPP)

2.2.4.5 Non-negative Matrix Factorization (NMF)

2.2.5 Nonlinear Manifold Learning

Linear methods such as PCA and MDS are convenient for projecting out-of-sample test points into the embedding space. However, they are unable to capture the structure of data that are sampled from nonlinear manifolds **CITE REFERENCE**. This section will discuss a variety of nonlinear dimensionality reduction and manifold learning approaches. All methods reviewed assume the data is distributed along a d-dimensional sub-manifold $\mathcal{X}$ embedded in $\mathbb{R}^D$.

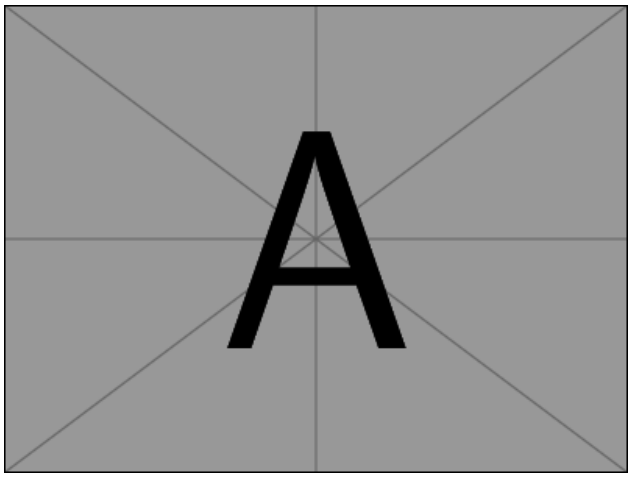


Figure 2-4: Example of a nonlinear manifold.

2.2.5.1 Graph-based Methods

Nonlinear dimensionality reduction methods typically rely on the use of computational graphs. These graphs represent data structure pooled from local neighborhoods of samples. *Spectral graph theory* focuses on constructing, analyzing and manipulating graphs. It has proved useful for object representation, graph visualization, spectral clustering, dimensionality reduction and numerous other applications in chemistry, physics, signal processing and computer science (Shuman et al., 2013; Bengoetxea, 2002). An overview of computational

graphs as well as prominent methods for graph construction in manifold learning are presented. Additionally, geodesic distance approximation from pairwise distances is reviewed.

Terminology. Many dimensionality reduction methods in the literature are interested in analyzing relationships between samples defined on an undirected, weighted graph $G = \{\mathcal{V}, \mathcal{E}, \boldsymbol{W}\}$, which consists of a finite set of *vertices* $\mathcal{V}$ (also called *nodes* or *points*) with cardinality $\mathcal{V} = N$, a set of *edges* $\mathcal{E} \subset \mathcal{V} \times \mathcal{V} = [\mathcal{V}]^2$ (also known as *arcs* or *lines*) and a weighted *adjacency* or *affinity* matrix $\boldsymbol{W}$ (Shuman et al., 2013; Livi and Rizzi, 2013; Bengoetxea, 2002). The the size or *order* of a graph is defined by the number of nodes $|\mathcal{V}|$ and edges $|\mathcal{E}|$. If two vertices in G, say $\boldsymbol{u}, \boldsymbol{v} \in \mathcal{V}$, are connected by an edge $e \in \mathcal{E}$, this is denoted by $e = (\boldsymbol{u}, \boldsymbol{v})$ and the two vertices are said to be *adjacent* or *neighbors*. When edges do not have a direction, they are coined as undirected. A graph solely containing this type of connection is termed as an *undirected graph*. When all edges have directions, meaning $(\boldsymbol{u}, \boldsymbol{v})$ and $(\boldsymbol{v}, \boldsymbol{u})$ are distinguishable, the graph is said to be *directed*. In the literature, the term *arc* is typically used to denote connections between nodes in directed graphs, while *edge* is used when they are undirected. The graph-based methods included in this literature review focus on analyzing affinities between data samples in undirected graphs. Moreover, a *path* between any two nodes in $\boldsymbol{u}, \boldsymbol{u'} \in \mathcal{V}$ is a non-empty sequence of k different vertices $< \boldsymbol{v}_0, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_k >$ where $\boldsymbol{u} = \boldsymbol{v}_0, \boldsymbol{u}' = \boldsymbol{v}_k$ and $(\boldsymbol{v}_{i-1}, \boldsymbol{v}_i) \in \mathcal{E}$, $i = 1, 2, \ldots, k$. Additionally, a graph is said to be *acyclic* if there are no cycles between its edges, regardless of whether it is directed or undirected.

When using graphs for dimensionality reduction, vertices usually represent features of individual samples, and edges express relationships between them. The most straight-forward way to construct a graph is to instantiate edges between every vertex in the graph, where each edge is weighted by the distance between the vertices it connects according to a pre-defined metric. This type of graph is called *full mesh*. Weights on edges are captured in the graph adjacency matrix $\boldsymbol{W}$. When weights are not naturally defined by an application, a common way to the define the weight of an edge connecting vertices $\boldsymbol{u} \sim \boldsymbol{u}'$ is by a symmetric affinity

function $W_{\boldsymbol{u},\boldsymbol{u}'} = K(\boldsymbol{u};\boldsymbol{u}')$; typically a *radial basis function (RBF)* or *heat kernel*, defined as:

$$\boldsymbol{W}_{\boldsymbol{u},\boldsymbol{u}'} = w_{\boldsymbol{u},\boldsymbol{u}'} = \exp\left(-\frac{||\boldsymbol{u}-\boldsymbol{u}'||^2}{\beta}\right) \tag{2-5}$$

where β is the non-negative *bandwidth* of the kernel. Vertices will have a nonzero weight only if they fall within the nonzero mapping domain of the kernel. Additionally, a threshold could be set to truncate the weights of neighbors far from individual samples.

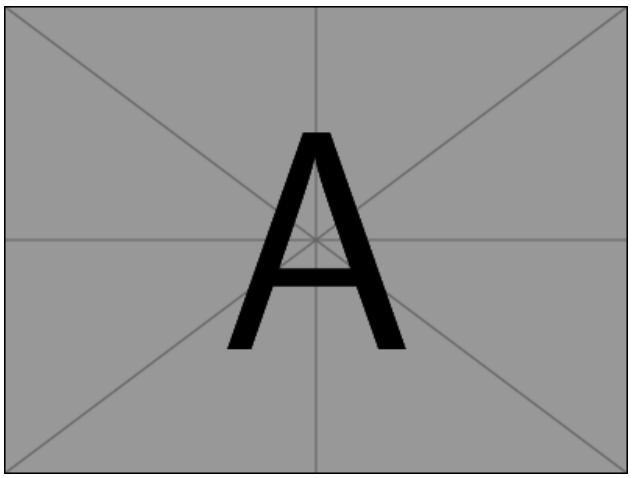


Figure 2-5: Examples of K-nearest neighbor and ϵ-ball graphs

***K*-Nearest Neighbor Graph**. In a K-nearest neighbor graph, every data point (vertex) $\boldsymbol{x}_n \in \boldsymbol{X}$ is connected by edges to its K-nearest neighbors, where $K \in \mathbb{Z}^+$ is fixed. An example of a K-nearest neighbor graph is depicted in a of Figure 2-5. The downside of this graph is that it might impose edges between neighbors that should not actually be connected, as in the case where a sample is metrically distant from all of its nearest neighbors. Although, this feature may actually be useful in domains such as outlier detection, where low adjacency weights indicate that the sample is far form the sampling distribution. Two alternative K-nearest neighbor graphs, a symmetric and mutual neighbors, might instead by utilized. In the symmetric K-nearest neighbors graph, two vertices $\boldsymbol{u}$ and $\boldsymbol{u}'$ if $\boldsymbol{u}$ is among the K-nearest neighbors of $\boldsymbol{u}'$ *or* $\boldsymbol{u}'$ is among the neighbors of $\boldsymbol{u}$. The mutual K-nearest neighbors graph, however, only connects vertices $(\boldsymbol{u},\boldsymbol{u}')$ if $\boldsymbol{u}$ is among the K-nearest neighbors of $\boldsymbol{u}'$ *and* $\boldsymbol{u}'$ is

among the K-nearest neighbors of $\boldsymbol{u}$. The weights on each edge are provided as the similarity of the adjacent nodes.

ϵ-Neighborhood Graph. Another method for graph construction is to use ϵ-neighborhoods (or ϵ-balls). In this graph, two vertices $(\boldsymbol{u}, \boldsymbol{u}')$ are connected by an edge if and only if the distance between them is equal to or smaller than some value ϵ, $\mathcal{D}_{\mathcal{U}}(\boldsymbol{u}, \boldsymbol{u}') \leq \epsilon$. This idea is represented in b of Figure 2-5. In both the K-nearest and ϵ-neighborhood graphs, a parameter controlling the number of edges in the graph, K or ϵ, must be chosen. These parameters are highly influential for graph construction and can thus greatly affect dimensionality reduction quality. Contrary to the K-nearest neighbor graph, an ϵ-neighborhood will not create connections between distant vertices. However, when the data is sampled sparsely from a highly-curved manifold, the ϵ-neighbor graph will not be able to appropriately capture the geometry (Thorstensen, 2009).

Geodesic Distance Approximation. The ultimate goal of manifold learning is to uncover an underlying low-dimensional sub-manifold which is embedded in $\mathbb{R}^D$ **CITE**. Many dimensionality reduction methods in the literature discover projections of data into a low-dimensional space which preserve topological ordering of the data **CITE**. These processes require a notion of distance between samples. *Euclidean distance* is a popular metric which captures the straight-line disparity between two points. As shown in Figure 2-6, however, samples that are actually distant on the manifold may appear deceptively close in the high-dimensional input feature space, as measured by Euclidean distance (Tenenbaum et al., 2000). *Geodesic distance*, also called *curvilinear* or *shortest-path distance*, Figure 2-6, on the other hand, follows the curvature of a manifold and may provide a better measure of dissimilarity between data samples. Geodesic distance can be estimated by the shortest path through a graph constructed by assuming the distances between neighbors is locally Euclidean **CITE**. This can be conceptualized by a simple example. The Earth is a sphere and naturally has curvature. Two people standing in a room, however, would estimate the distance between themselves by a straight line. Thus, in a very local region on the Earth, the measure of

curvature would be negligible and the true distances between objects could be estimated with Euclidean distance. The same concept is true for manifolds where, if data is sampled densely enough, geodesic distance can be approximated by the shortest-path through a neighborhood graph where the dissimilarities between neighbors is assumed to be locally Euclidean. Geodesic distance can be estimated efficiently by methods such as Dijskstra's or Floyd's shortest-path algorithms (Tenenbaum et al., 2000).

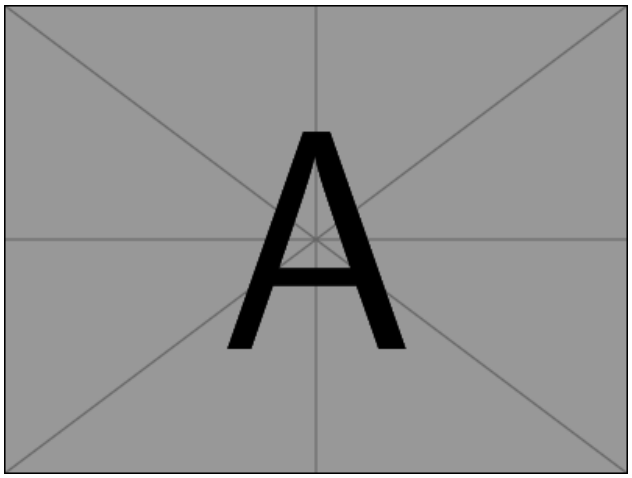


Figure 2-6: Demonstration of geodesic distance

2.2.5.2 General Graph Embedding Framework

2.2.5.3 Isomap

While MDS has proven to be successful in a variety of applications, it suffers from the fact that is solely aims to retain pairwise Euclidean distances and does not consider the distributions of neighboring samples. This implies that MDS is not able to capture the geometry of high-dimensional data which lies on or near to a curved manifold, such as the Swiss roll dataset van der Maaten et al. (2007); Chao et al. (2019). Isometric Feature Mapping (Isomap) (Tenenbaum et al., 2000) is a technique which resolves this problem by attempting to preserve pairwise geodesic distances between datapoints. Isomap can be considered as a generalization of classical MDS in which the pairwise distance matrix is

replaced by a matrix of pairwise geodesic distances approximated by distances in the graph (Thorstensen, 2009). The classic, unsupervised algorithm consists of a few steps:

1. Given a set of input data $\boldsymbol{X} = \{\boldsymbol{x}_n\}_{n=1}^N \subset \mathbb{R}^D$, construct a sparse neighborhood graph (such as the K-nearest or ϵ-ball graphs discussed previously) where each edge is weighted by the Euclidean distance between the neighbors it connects:

$$\boldsymbol{W}_{mn} = w_{mn} = ||\boldsymbol{x}_m - \boldsymbol{x}_n||^2 \tag{2-6}$$

   where $\boldsymbol{W}$ is the graph adjacency matrix.

2. Next, the geodesic distances between all pairs of samples is computed by finding the shortest paths between the points through the graph. This is commonly done with Dijkstra's or Flyod's shortest-path algorithms **CITE**.

3. These geodesic distances form a pairwise distance matrix which is substituted into classical MDS as described in Section 2.2.4.2. This provides the low-dimensional embedding coordinates $\boldsymbol{Z} = [\boldsymbol{z}_1^T, \ldots, \boldsymbol{z}_N^T] \in \mathbb{R}^{N \times d}$ of high-dimensional input data $\boldsymbol{X} = [\boldsymbol{x}_1^T, \ldots, \boldsymbol{x}_N^T] \in \mathbb{R}^{N \times D}$, where $d \ll D$.

While Isomap has been successfully applied in the areas of **FILL IN APPLICATIONS**, a few important weaknesses are prevalent. First, Isomap may be topologically unstable. That is, it may construct erroneous connections in the neighborhood graph. This is known as short-circuiting, and it can severely impair the performance of Isomap. Several approaches have been proposed to nullify the short-circuiting problem, such as removing datapoints with large total flows or by removing nearest neighbors that violate local linearity of the neighborhood graph (van der Maaten et al., 2007). Another weakness of Isomap is that it may not perform correctly if there are holes in the manifold, as this causes the geodesic distances of some samples to appear further on the manifold than they truly are. A third weakness is that Isomap can fail if the manifold is non-convex. Therefore, we see that Isomap can perform very well due to theoretical guarantees on qualities such as convergence, as long as the manifold is isometric to a convex open set of $\mathbb{R}^d$, $\mathcal{D}_{\mathcal{X}}(\boldsymbol{u}, \boldsymbol{u}') = \mathcal{D}_{\mathcal{Y}}(f(\boldsymbol{u}), f(\boldsymbol{u}'))$, meaning that the geodesic distances in the graph are almost equal to the Euclidean distances in the embedding space $\mathbb{R}^d$. Continuing, an additional drawback of Isomap is the fact that it requires the decomposition of a large, dense Gram matrix which scales with the number of training data points. If the

dataset grows too large, a solution will no longer be tractable. Furthermore, the constraint on $\mathcal{X}$ to be isometric to a convex open set of $\mathbb{R}^d$ is rarely met. As mentioned in (Thorstensen, 2009), these problems may be circumvented by sparsifying large datasets using landmarks, as with Landmark Isomap (**CITE Silva 2003**) and looking at conformal maps, as is done in Conformal Isomap **CITE de Silva 2002** Finally, as with most nonlinear manifold learning techniques, it is nontrivial to embed out-of-sample data points into the lower dimensional feature space.

As with most traditional manifold learning methods in the literature, Isomap is not inherently well-suited for classification tasks. However, supervised approaches which consider class label information have been adopted to increase class separability in the latent embedding space.

2.2.5.4 Locally Linear Embedding (LLE)

2.2.5.5 Laplacian Eigenmaps (LE)

Classical LE. Similar to LLE, Laplacian Eigenmaps, or *Spectral Embedding*, is a nonlinear dimensionality reduction technique which aims to preserve local structure of data (Raducanu and Dornaika, 2012; van der Maaten et al., 2007). Using *spectral graph theory*, LE computes low-dimensional representations of data in which the dissimilarities between datapoints and their neighbors (according to an affinity measure) are minimized. The name *Laplacian Eigenmaps* is derived by the use of Laplacian regularization in the optimization procedure (Thorstensen, 2009). Given a set of N samples $\boldsymbol{X} = \{\boldsymbol{x}_n\}_{n=1}^N \subset \mathbb{R}^D$, the first step of LE is to define a *neighborhood graph on the samples*. This graph, also called an *affinity* or *adjacency* matrix can be constructed in a variety of ways, such as K-nearest neighbor, ϵ-ball, full mesh, or by weighting each edge $\boldsymbol{x}_m \sim \boldsymbol{x}_n$ by a symmetric affinity function $W_{mn} = K(\boldsymbol{x}_m; \boldsymbol{x}_n)$, typically a radial basis or heat kernel:

$$\boldsymbol{W}_{mn} = w_{mn} = \exp\left(-\frac{||\boldsymbol{x}_m - \boldsymbol{x}_n||^2}{\beta}\right) \tag{2-7}$$

where the kernel bandwidth β is typically set as the variance of the dataset (Raducanu and Dornaika, 2012; Thorstensen, 2009).

The goal is to uncover the latent data representations $\{\boldsymbol{z}_n\}_{n=1}^N \subset \mathbb{R}^d$ where $d \ll D$ which minimizes the objective

$$J(\boldsymbol{W}, \boldsymbol{Z}) = \frac{1}{2} \sum_{m,n} ||\boldsymbol{z}_m - \boldsymbol{z}_n||^2 w_{mn} = tr(\boldsymbol{Z}^T \boldsymbol{L} \boldsymbol{Z}) \tag{2-8}$$

with $\boldsymbol{W}$ denoting the symmetric affinity matrix, $\boldsymbol{D}$ the diagonal weight matrix whose entries are the sum of the rows (or columns since $\boldsymbol{W}$ is symmetric) of $\boldsymbol{W}$ (i.e. $d_{mm} = \sum_n w_{mn}$, and is 0 otherwise). The graph Laplacian matrix is provided as $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$. The matrix $\boldsymbol{Z} = [\boldsymbol{z}_1^T, \ldots, \boldsymbol{z}_N^T]$ is the $N \times d$ embedding matrix and $tr(.)$ denotes the trace of a matrix. The n^{th} row of matrix $\boldsymbol{Z}$ provides the vector $\boldsymbol{z}_n$, which is the latent representation of sample $\boldsymbol{x}_n$. This objective discourages projecting similar points in the input feature space to disparate regions of the embedding space by enforcing heavy penalization.

The latent sample coordinates $\boldsymbol{Z}$ are found as the solution to the optimization problem:

$$\min_{\boldsymbol{Z}} tr(\boldsymbol{Z}^T \boldsymbol{L} \boldsymbol{Z}) \quad s.t. \quad \boldsymbol{Z}^T \boldsymbol{D} \boldsymbol{Z} = \boldsymbol{I}, \boldsymbol{Z}^T \boldsymbol{L} \boldsymbol{e} = \boldsymbol{0} \tag{2-9}$$

where $\boldsymbol{I}$ is the identity matrix and $\boldsymbol{e} = (1, \ldots, 1)^T$. The first constraint eliminates the trivial solution $\boldsymbol{Z} = \boldsymbol{0}$ (scaling) and the second constraint avoids the trivial solution $\boldsymbol{Z} = \boldsymbol{e}$ (uniqueness). By applying the Langrange multiplier method and using the fact that $\boldsymbol{L}\boldsymbol{e} = \boldsymbol{0}$, the low-dimensional data representations can be found by solving the generalized eigenvalue problem:

$$\boldsymbol{L}\boldsymbol{v} = \lambda \boldsymbol{D}\boldsymbol{v} \tag{2-10}$$

The column vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_N$ are the solutions of Equation 2-10, ordered to the corresponding eigenvalues, in ascending order, $\lambda_1 = 0 \leq \lambda_2 \leq \cdots \leq \lambda_N$. The embedding of the input samples given by the matrix $\boldsymbol{Z}$, is obtained by concatenating the eigenvectors of the d smallest non-zero eigenvalues. $\boldsymbol{Z}$ is a $N \times d$ matrix, where $d < N$ is the dimensionality of the embedded

space. From observation, it is clear that the embedding dimensionality is limited by the number of samples N.

Supervised LE (S-LE). In order to adopt LE for classification,Raducanu and Dornaika (Raducanu and Dornaika, 2012) proposed a supervised LE which minimizes the margin between samples with similar class labels and maximizes the margin between samples with opposing class labels. Supervised LE utilizes discriminative information contained in the class labels when finding the nonlinear embedding (spectral projection).

In order to discover both geometrical and discriminative manifold structure, supervised LE splits the global graph into two components: the within-class graph G_w and the between-class graph G_b. To define the margin, they define two subsets, $N_w(\boldsymbol{x}_n)$ and $N_b(\boldsymbol{x}_n)$ for each sample $\boldsymbol{x}_n$. These two subsets contain the neighbors of $\boldsymbol{x}_n$ sharing the same label and having different labels, respectively, which have a similarity higher than the average.

$$N_w(\boldsymbol{x}_n) = \{\boldsymbol{x}_m | l_m = l_n, \exp\left(-\frac{||\boldsymbol{x}_n - \boldsymbol{x}_m||^2}{\beta}\right) > AS(\boldsymbol{x}_n)\} \tag{2-11}$$

$$N_b(\boldsymbol{x}_n) = \{\boldsymbol{x}_m | l_m \neq l_n, \exp\left(-\frac{||\boldsymbol{x}_n - \boldsymbol{x}_m||^2}{\beta}\right) > AS(\boldsymbol{x}_n)\} \tag{2-12}$$

where $AS(\boldsymbol{x}_n) = \frac{1}{N}\sum_{n=1}^{N} \exp\left(-\frac{||\boldsymbol{x}_n - \boldsymbol{x}_m||^2}{\beta}\right)$ denotes the average similarity of the sample $\boldsymbol{x}_n$ to the rest of the data. From Equations 2-11 and 2-12 it is clear that the neighborhoods for each data sample are not necessarily the same size. As a result, this function constructs the affinity graph according to both the local density and similarity between data samples in the input feature space.

With the two sets defined, the within-class and between-class weight matrices $\boldsymbol{W}_w$ and $\boldsymbol{W}_b$ are formed from the adjacency graphs G_w and G_b, respectively. These weight matrices are

defined as:

$$W_{w,mn} = \begin{cases} \exp\left(-\frac{||\boldsymbol{x}_n - \boldsymbol{x}_m||^2}{\beta}\right), & \text{if } \boldsymbol{x}_n \in N_w(\boldsymbol{x}_m) \text{ or } \boldsymbol{x}_m \in N_w(\boldsymbol{x}_n) \\ 0, & \text{otherwise} \end{cases} \tag{2-13}$$

$$W_{b,mn} = \begin{cases} 1, & \text{if } \boldsymbol{x}_n \in N_b(\boldsymbol{x}_m) \text{ or } \boldsymbol{x}_m \in N_b(\boldsymbol{x}_n) \\ 0, & \text{otherwise} \end{cases} \tag{2-14}$$

and the global affinity matrix, $\boldsymbol{W}$, can be written as:

$$\boldsymbol{W} = \boldsymbol{W}_w + \boldsymbol{W}_b \tag{2-15}$$

In order to obtain the low-dimensional representations $\boldsymbol{z}_n$ of the input data $\boldsymbol{x}_n$, the following objective functions can be optimized for $\boldsymbol{Z}$:

$$\min \frac{1}{2} \sum_{m,n} ||\boldsymbol{z}_m - \boldsymbol{z}_n||^2 W_{w,mn} = tr(\boldsymbol{Z}^T \boldsymbol{L}_w \boldsymbol{Z}) \tag{2-16}$$

$$\max \frac{1}{2} \sum_{m,n} ||\boldsymbol{z}_m - \boldsymbol{z}_n||^2 W_{b,mn} = tr(\boldsymbol{Z}^T \boldsymbol{L}_b \boldsymbol{Z}) \tag{2-17}$$

where $\boldsymbol{L}_w = \boldsymbol{D}_w - \boldsymbol{W}_w$ and $\boldsymbol{L}_b = \boldsymbol{D}_b - \boldsymbol{W}_b$ indicate the corresponding graph Laplacians of the within-class and between-class affinity graphs, respectively. The matrix $\boldsymbol{Z} = [\boldsymbol{z}_1^T, \ldots, \boldsymbol{z}_N^T]$ contains the low-dimensional representations of the input samples in its rows.

By merging the two objective functions, the final optimization problem is formulated as:

$$\arg\max_{\boldsymbol{Z}} \left\{ \gamma tr(\boldsymbol{Z}^T \boldsymbol{L}_b \boldsymbol{Z}) + (1-\gamma) tr(\boldsymbol{Z}^T \boldsymbol{W}_w \boldsymbol{Z}) \right\} \quad s.t. \quad \boldsymbol{Z}^T \boldsymbol{D}_w \boldsymbol{Z} = \boldsymbol{I} \tag{2-18}$$

The term γ is a scalar value in $[0, 1]$ which determines the trade-off between pulling similar samples toward each other in the latent space and pushing heterogeneous points away. A value of $\gamma = 1$ forces the objective to solely focus on maximizing the margin between dissimilar points. Alternatively, a value of $\gamma = 0$ priorities the objective on embedding homogeneous

samples in close spatial proximity. By defining matrix $\boldsymbol{B} = \gamma \boldsymbol{L}_b + (1-\gamma)\boldsymbol{W}_w$, the problem becomes:

$$\arg\max_{\boldsymbol{Z}} \left(\boldsymbol{Z}^T \boldsymbol{B} \boldsymbol{Z}\right) \quad s.t. \quad \boldsymbol{Z}^T \boldsymbol{D}_w \boldsymbol{Z} = \boldsymbol{I} \tag{2-19}$$

The low-dimensional embedding matrix $\boldsymbol{Z}$ can be found by solving the generalized eigenvalue problem:

$$\boldsymbol{B}\boldsymbol{v} = \lambda \boldsymbol{D}_w \boldsymbol{v} \tag{2-20}$$

The column vectors $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_N$ are the generalized eigenvectors of Equation 2-20 arranged by descending eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_d$. Then the $N \times d$ embedding matrix $\boldsymbol{Z} = [\boldsymbol{z}_1^T, \ldots, \boldsymbol{z}_N^T]$ is provided by concatenating the obtained eigenvectors $\boldsymbol{Z} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_d]$.

The primary difference between the classic LE and S-LE is that traditional LE solely attempts to preserve the spatial relationships between samples, and thus, does not consider label information when learning the embeddings. Alternatively, S-LE aims at aiding discriminant analysis by collapsing the distance between samples with the same label that are in close spatial proximity and pushing away spatial neighbors with differing class labels. This is done through the utilization of two affinity graphs: the within-class and between-class graphs. As with most graph-based methods, LE results vary highly according to the choice of neighborhood size. However, choosing the size of K or ϵ in advance can be very difficult. S-LE does not require user-defined graph parameters, other than those associated with the chosen affinity measure. Instead, graph edges are chosen according to an adaptive neighborhood for each sample. Both methods, however, suffer from inherent difficulties associated with nonlinear manifold learning, namely, selecting the intrinsic embedding dimensionality and handling out-of-sample extensions.

Despite these nuances, LE (and its variants) have been successfully applied in nonlinear dimensionality reduction tasks for facial recognition, spectral clustering and object classification van der Maaten et al. (2007).

Apart from S-LE, other methods have been explored to integrate label information into Laplacian Eigenmaps. A review of supervised dimensionality reduction methods by Chao et al. (Chao et al., 2019) explains that author's have optimized the affinity matrix using label information after constructing from spatial proximity, proposed deep learning-based approaches to achieve supervised LE and integrated label information into the affinity matrix construction process.

The special feature exhibited by all Laplacian Eigenmap methods is the use of laplacian regularization, which enforces properties such as smoothness and provides a level of resistance toward the influences of outliers. This useful feature has been applied in a variety of supervised and semi-supervised tasks, such as hyperspectral and synthetic aperture radar remote sensing classification (Ratle et al., 2010; Ren et al., 2017), classification of synthetic data (Tsang and Kwok, 2007), zero-shot learning (Meng and Zhan, 2018) and reinforcement learning (Li et al., 2015).

2.2.5.6 Hessian Eigenmaps

2.2.5.7 Diffusion Maps

2.2.5.8 Sammon Mapping

2.2.5.9 Maximum Variance Unfolding (MVU)

2.2.6 Latent Variable Models

2.2.6.1 General Latent Variable Model (GLVM)

Factor Analysis (FA).

Probabilistic PCA (PPCA). PCA can also be analyzed from the viewpoint of factor analysis (FA), which is discussed later in this literature review. The basic idea, however, is that data observations $\boldsymbol{x}_n \in \mathbb{R}^D$ are realizations of a probability distribution with a prior on the lower-dimensional latent variable $\boldsymbol{z}_n \in \mathbb{R}^d$. A typical choice on this model is a Gaussian-Gaussian conjugate prior pair where the mean of the data likelihood is a linear function of the latent inputs. As an example, the prior over the hidden data representations can be expressed as Gaussian distribution

$$p(\boldsymbol{z}_n) = \mathcal{N}(\boldsymbol{z}_n|\boldsymbol{\mu}_0, \boldsymbol{S}_0) \tag{2-21}$$

and the data likelihood is denoted as a multivariate Gaussian

$$p(\boldsymbol{x}_n|\boldsymbol{z}_n, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{W}\boldsymbol{z}_n + \boldsymbol{\mu}, \boldsymbol{\Psi}) \tag{2-22}$$

where $\boldsymbol{W}$ is a $D \times d$ *factor loading matrix*, and $\boldsymbol{\Psi}$ is a $D \times D$ covariance matrix. The objective of FA is to compute the posterior over the latent factors in hopes that they will reveal something interesting about the data (Murphy, 2012). Classical PCA assumes the data covariance to be $\boldsymbol{\Psi} = \sigma^2\boldsymbol{I}$ with $\sigma^2 \to 0$, thus the model is deterministic. Alternatively, when $\sigma^2 > 0$,the projection is no longer orthogonal since it is shrunk toward the prior mean. The trade-off is that the reconstructions of $\boldsymbol{x}_n$ with be closer to the data mean.

While the typical approach for fitting PCA is to use the method of eigenvectors of Singular Value Decomposition (SVD), it can also be fit with Expectation-Maximization (EM). This formulation may be more computationally efficient for high-dimensional data. By using Gaussian Processes, PPCA may also be extended to learn nonlinear mappings between the input and latent feature spaces (van der Maaten et al., 2007).

Supervised PCA.

Supervision has also been applied to PCA. Two such examples, Supervised PCA and Discriminative Supervised PCA, were provided by Murphy in (Murphy, 2012). The two are briefly described.

Supervised PCA (Latent Factor Regression) *Supervised PCA* or *Bayesian factor regression* is a model like PCA, except that the target variable (or label), l_n is taken into account when learning the low-dimensional embedding. For the case of binary classification,

the Bayesian model can be decomposed into the following elements:

$$p(\boldsymbol{z}_n) = \mathcal{N}(0, \boldsymbol{I}_d) \tag{2-23}$$

$$p(l_n|\boldsymbol{z}_n) = \text{Ber}(\text{sigm}(\boldsymbol{w}_l^T \boldsymbol{z}_n)) \tag{2-24}$$

$$p(\boldsymbol{x}_n|\boldsymbol{z}_n) = \mathcal{N}(\boldsymbol{W}_x \boldsymbol{z}_n + \boldsymbol{\mu}_x, \sigma^2 \boldsymbol{I}_D) \tag{2-25}$$

Discriminative Supervised PCA

2.2.6.2 Generative Topographic Mapping (GTM)

2.2.7 Competitive Hebbian Learning

2.2.8 Deep Learning

2.2.9 Current State of the Art

2.2.10 UMAP

2.2.11 Stochastic Neighbor Embedding (SNE and t-SNE)

2.2.12 NCA

2.2.13 Multiple Instance Learning on Manifolds

2.3 Metric Embedding

The concepts of "near" and "far" are very powerful and useful utilities in everyday life. They classify the relationship between two "primitives" as being similar or dissimilar, as well as the degree of compatibility (Thorstensen, 2009). As an example, a medical doctor might consider a machine learning researcher and a software engineer as being similar (near), because they both perform research for computer applications. However, the same researcher and engineer would likely consider their jobs as being very disparate (far) based on the details of their work. In order to capture this abstraction of distance, a *metric space* is defined as a mathematical construction of this vague generality.

Definition 2.3.1. Metric Space A *metric space* is an ordered pair $(\mathcal{X}, \mathcal{D})$ where $\mathcal{X}$ is a set and $\mathcal{D}$ is a metric on $\mathcal{X}$, or $\mathcal{D} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ such that $\forall \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z} \in \mathcal{X}$, the following holds:

1. Non-negativity: $\mathcal{D}(\boldsymbol{x}, \boldsymbol{y}) \geq 0$
2. Identity: $\mathcal{D}(\boldsymbol{x}, \boldsymbol{y}) = 0 \iff \boldsymbol{x} = \boldsymbol{y}$
3. Symmetry: $\mathcal{D}(\boldsymbol{x}, \boldsymbol{y}) = \mathcal{D}(\boldsymbol{y}, \boldsymbol{x})$
4. Triangle Inequality: $\mathcal{D}(\boldsymbol{x}, \boldsymbol{z}) \leq \mathcal{D}(\boldsymbol{x}, \boldsymbol{y}) + \mathcal{D}(\boldsymbol{y}, \boldsymbol{z})$

Explain more about these properties...

The goal of *metric embedding learning* it to learn a function $f_\theta(\boldsymbol{x}) : \mathbb{R}^D \to \mathbb{R}^d$ which maps semantically similar points from the data input feature space of $\mathbb{R}^D$ onto *metrically close* points in $\mathbb{R}^d$. Similarly, f_θ should map semantically different points in $\mathbb{R}^D$ onto metrically distant points in $\mathbb{R}^d$. The function f_θ is parameterized by θ and can be anything ranging from a linear transformation to a complex non-linear mapping as in the case of deep artificial neural networks (an Lucas Beyer and Leibe, 2017). Let $\mathcal{D}(\boldsymbol{x}, \boldsymbol{y}) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ be a metric function measuring similarity or dissimilarity in the embedded space. For succinctness, $\mathcal{D}_{i,j} = \mathcal{D}(f_\theta(\boldsymbol{x_i}), f_\theta(\boldsymbol{x_j}))$ defines the dissimilarity between samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, after being embedded.

2.3.0.1 Metric Learning

Thorstensen thesis, Raducanu S-LE

2.3.1 Ranking Loss

2.3.1.1 Pairwise Loss

2.3.1.2 Contrastive Loss

Definition of Contrastive Loss:

2.3.1.3 Triplet Loss

Definition of Triplet Loss:

Triplet loss was extended in (Sohn, 2016) to simultaneously optimize against N negative classes.

2.3.1.4 Large-Margin K-Nearest Neighbors (LMNN)

2.3.1.5 FaceNet

FaceNet is a convolutional neural network which learns a mapping from face images to a compact Euclidean space where distances directly correspond to a measure of face similarity (Schroff et al., 2015).

$$||f(x_i^a) - f(x_i^p)||_2^2 + \alpha < ||f(x_i^a) - f(x_i^n)||_2^2, \quad \forall (f(x_i^a), f(x_i^p), f(x_i^n)) \in \mathcal{T} \tag{2-26}$$

$$\mathcal{L} = ||f(x_i^a) - f(x_i^p)||_2^2 - ||f(x_i^a) - f(x_i^n)||_2^2 + \alpha \tag{2-27}$$

2.3.1.6 Siamese Neural Networks

2.3.2 Manifold Regularization

2.3.3 Multiple Instance Metric Learning

CHAPTER 3
PROBLEM DESCRIPTION

CHAPTER 4
EXPERIMENTAL DESIGN

CHAPTER 5
PRELIMINARY WORK

CHAPTER 6
FUTURE TASKS

CHAPTER 7
CONCLUSIONS

REFERENCES

Alexander Hermans an Lucas Beyer and Bastian Leibe. In defense of the triplet loss for person re-identification. *CoRR*, abs/1703.07737, 2017. URL http://arxiv.org/abs/1703.07737.

M. Belkin and P. Niyogi. Semi-supervised learning on riemannian manifolds. *Machine Learning*, 56(1):209–239, Jul 2004. ISSN 1573-0565. doi: 10.1023/B:MACH.0000033120.25363.1e.

M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *J. Mach. Learn. Res.*, 7:2399–2434, December 2006. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=1248547.1248632.

Y. Bengio, A. C. Courville, and P. Vincent. Unsupervised feature learning and deep learning: A review and new perspectives. *CoRR*, abs/1206.5538, 2012. URL http://arxiv.org/abs/1206.5538.

E. Bengoetxea. *Inexact Graph Matching Using Estimation of Distribution Algorithms*. PhD thesis, Ecole Nationale Supérieure des Télécommunications, Paris, France, Dec. 2002.

C. Bishop, M. Svensn, and C. K. I. Williams. Gtm: The generative topographic mapping. 10: 215–234, January 1998. URL https://www.microsoft.com/en-us/research/publication/gtm-the-generative-topographic-mapping/.

J. Bocinsky. Learning multiple target concepts from uncertain, ambiguous data using the adaptive cosine estimator and spectral match filter. Master's thesis, Univ. of Florida, Gainesville, FL, May 2019.

M. Carbonneau, V. Cheplygina, E. Granger, and G. Gagnon. Multiple instance learning: A survey of problem characteristics and applications. *CoRR*, abs/1612.03365, 2016. URL http://arxiv.org/abs/1612.03365.

Jing Chai, Xinghao Ding, Hongtao Chen, and Tingyu Li. Multiple-instance discriminant analysis. *Pattern Recognition*, 47(7):2517 – 2531, 2014. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2014.02.002". URL http://www.sciencedirect.com/science/article/pii/S0031320314000387.

G. Chao, Y. Luo, and W. Ding. Recent advances in supervised dimension reduction: A survey. *Machine Learning and Knowledge Extraction*, 1(1):341–358, 2019. ISSN 2504-4990. doi: 10.3390/make1010020.

B. Chaudhuri and S. Parui. Target detection: Remote sensing techniques for defence applications. *Defence Science Journal*, 45:285–291, 04 1995. doi: 10.14429/dsj.45.4135.

M. Cook. Task driven extended functions of multiple instances (td-efumi). Master's thesis, Univ. of Missouri, Columbia, MO, 2015.

T. G. Dietterich, R. H. Lathrop, and T. Lozano-Prez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1):31 – 71, 1997. ISSN 0004-3702. doi: https://doi.org/10.1016/S0004-3702(96)00034-3.

Defense Systems Information Analysis Center DSIAC. DSIAC MS-003-DB Algorithm Development Database. https://www.dsiac.org/resources/research-materials/cds-dvds-databases-digital-files/atr-algorithm-development-image, 2014.

X. Du. *Multiple Instance Choquet Integral For MultiResolution Sensor Fusion*. PhD thesis, Univ. of Missouri, Columbia, MO, Dec. 2017.

X. Du and A. Zare. Technical report: Scene label ground truth map for muufl gulfport data set. Technical Report 417, University of Florida, Gainesville, FL, April 2017. URL http://ufdc.ufl.edu/IR00009711/00001.

P. Gader, A. Zare, R. Close, J. Aitken, and G. Tuell. Muufl gulfport hyperspectral and lidar airborne data set. Technical Report 570, University of Florida, Gainesville, FL, October 2013.

Xiurui Geng, Luyan Ji, and Yongchao Zhao. The basic equation for target detection in remote sensing, 2017.

T. Glenn, A. Zare, P. Gader, and D. Dranishnikov. Bullwinkle: Scoring code for sub-pixel targets. URL https://github.com/GatorSense/MUUFLGulfport/.

Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

Balzs Kegl, Donald Wunsch, and Andrei Zinovyev. *Principal Manifolds for Data Visualisation and Dimension Reduction, LNCSE 58*. 01 2008. ISBN 978-3-540-73750-6.

T. Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480, Sep. 1990. ISSN 0018-9219. doi: 10.1109/5.58325.

H. Li, D. Liu, and D. Wang. Approximate policy iteration with unsupervised feature learning based on manifold regularization. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pages 1–6, July 2015.

L. Livi and A. Rizzi. The graph matching problem. *Pattern Anal. Appl.*, 16(3):253–283, Aug 2013. ISSN 1433-7541. doi: 10.1007/s10044-012-0284-8.

M. Meng and X. Zhan. Zero-shot learning via low-rank-representation based manifold regularization. *IEEE Signal Processing Letters*, 25(9):1379–1383, Sep. 2018. ISSN 1070-9908. doi: 10.1109/LSP.2018.2857201.

K. P. Murphy. *Machine Learning: A Probabilistic Perspective*. The MIT Press, 2012. ISBN 0262018020, 9780262018029.

M. Nickel and D. Kiela. Poincaré embeddings for learning hierarchical representations. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 6338–6347. Curran Associates, Inc., 2017.

E. J. Palomo and E. Lopez-Rubio. The growing hierarchical neural gas self-organizing neural network. *IEEE Transactions on Neural Networks and Learning Systems*, 28(9):2000–2009, Sep. 2017. ISSN 2162-237X. doi: 10.1109/TNNLS.2016.2570124.

B. Raducanu and F. Dornaika. A supervised non-linear dimensionality reduction approach for manifold learning. *Pattern Recognition*, 45(6):2432 – 2444, 2012. ISSN 0031-3203.

F. Ratle, G. Camps-Valls, and J. Weston. Semisupervised neural networks for efficient hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 48 (5):2271–2282, May 2010. ISSN 0196-2892. doi: 10.1109/TGRS.2009.2037898.

Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *CoRR*, abs/1804.02767, 2018. URL http://arxiv.org/abs/1804.02767.

B. Ren, B. Hou, J. Zhao, and L. Jiao. Unsupervised classification of polarimetirc sar image via improved manifold regularized low-rank representation with multiple features. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(2):580–595, Feb 2017. ISSN 1939-1404. doi: 10.1109/JSTARS.2016.2573380.

I. Rish, G. Grabarnik, G. A. Cecchi, F. Pereira, and G. J. Gordon. Closed-form supervised dimensionality reduction with generalized linear models. pages 832–839, 01 2008. doi: 10.1145/1390156.1390261.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. *CoRR*, abs/1503.03832, 2015. URL http://arxiv.org/abs/1503.03832.

D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Processing Magazine*, 30(3):83–98, May 2013.

Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 1857–1865. Curran Associates, Inc., 2016.

Yu-Yin Sun, Michael K. Ng, and Zhi-Hua Shou. Multi-instance dimensionality reduction. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, AAAI'10, pages 587–592. AAAI Press, 2010. URL http://dl.acm.org/citation.cfm?id=2898607.2898702.

R. Talmon, S. Mallat, H. Zaveri, and R. R. Coifman. Manifold learning for latent variable inference in dynamical systems. *IEEE Transactions on Signal Processing*, 63(15):3843–3856, Aug 2015. ISSN 1053-587X. doi: 10.1109/TSP.2015.2432731.

J. B. Tenenbaum, V. Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000. ISSN 0036-8075. doi: 10.

1126/science.290.5500.2319. URL https://science.sciencemag.org/content/290/5500/2319.

S. Theodoridis and K. Koutroumbas. Kernel pca. In *Pattern Recognition, Fourth Edition*, chapter 6, pages 351–353. Academic Press, Inc., Orlando, FL, USA, 4th edition, 2008. ISBN 1597492728, 9781597492720.

N. Thorstensen. *Manifold learning and applications to shape and image processing*. PhD thesis, Ecole Nationale des Ponts et Chaussees, Paris, France, Nov. 2009.

I. W. Tsang and J. T. Kwok. Large-scale sparsified manifold regularization. In B. Schölkopf, J. C. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 1401–1408. MIT Press, 2007. URL http://papers.nips.cc/paper/3005-large-scale-sparsified-manifold-regularization.pdf.

L. van der Maaten, E. Postma, and H. Herik. Dimensionality reduction: A comparative review. *Journal of Machine Learning Research - JMLR*, 10, 01 2007.

E. Vural and C. Guillemot. A study of the classification of low-dimensional data with supervised manifold learning. *CoRR*, abs/1507.05880, 2018. URL http://arxiv.org/abs/1507.05880.

David Weinberger. Ai outside in: Machine learning's triangle of error. URL https://accelerate.withgoogle.com/stories/ai-outside.

X. Geng, D. Zhan, and Z. Zhou. Supervised nonlinear dimensionality reduction for visualization and classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 35(6):1098–1107, Dec 2005. ISSN 1083-4419.

Y. Xu, W. Ping, and A. T. Campbell.

A. Zare. *Hyperspectral Endmember Detection and Band Selection Using Bayesian Methods*. PhD thesis, Univ. of Florida, Gainesville, FL, 2008.

Alina Zare, Changzhe Jiao, and Taylor Glenn. Discriminative multiple instance hyperspectral target characterization. *IEEE Trans. Pattern Anal. Mach. Inteli.*, 40(10):2342–2354, Oct. 2018. doi: 10.1109/TPAMI.2017.2756632.

Hong Zhu, Li-Zhi liao, and Michael K. Ng. Multi-instance dimensionality reduction via sparsity and orthogonality. *Neural Comput.*, 30(12):3281–3308, dec 2018. ISSN 0899-7667. doi: 10.1162/neco_a_01140. URL https://doi.org/10.1162/neco_a_01140.