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

LIST OF FIGURES

LIST OF ABBREVIATIONS

APR	Axis-Parallel Rectangles
BS	Bag-Space Paradigm
CLFDA	Citation Local Fisher Linear Discriminant Analysis
CT	Computed Tomography
CHL	Competitive Hebbian Learning
DD	Diverse Density
DR	Dimensionality Reduction
DSIAC	Defense Systems Information Analysis Center
EM	Expectation-Maximization
EM-DD	Expectation-Maximization Diverse Density
ES	Embedded-Space Paradigm
FA	Factor Analysis
FP	False Positive
FPS	Frames per Second
GLVM	General Latent Variable Model
GTM	Generative Topographic Mapping
GPS	Global Positioning System
HAC	Hierarchical Agglomerative Clustering
HS	Hyperspectral
HSI	Hyperspectral Image
IID	Independently and Identically Distributed
iPALM	Intertial Proximal Alternating Linearized Minimization
IS	Instance-Space Paradigm
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
LFDA	Local Fisher Discriminant Analysis
LPP	Locality Preserving Projections
MDS	Multi-dimensional Scaling
MI	Multiple Instance
MI-ALM	Multiple Instance Augmented Lagrangian Multiplier
MIC	Multiple Instance Classification
MIDA	Multiple-Instance Discriminant Analysis
MidLABS	Multi-Instance Dimensionality reduction by Learning a mAximum Bag margin Subspace
MI-FEAR	Multiple-Instance Feature Ranking
MIL	Multiple Instance Learning
MIDR	Multiple Instance Dimensionality Reduction

MILES	Multiple-Instance Learning via Embedded Instance Selection
MLE	Maximum Likelihood Estimation
MWIR	Mid-wave Infrared
NCA	Neighborhood Component Analysis
PCA	Principal Component Analysis
PGM	Probabilistic Graphical Model
RBF	Radial Basis Function
ROI	Region of Interest
S-LE	Supervised Laplacian Eigenmaps
SMI	Standard Multiple Instance
SOA	State-of-the-Art
SOM	Self-organizing Feature Map
SVD	Singular Value Decomposition
SVM	Support Vector Machine
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
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 involve learning a labeling function which can differentiate between classes at test (Geng et al., 2017; Chaudhuri and Parui, 1995). 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 (Xu et al., 2014). 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 (Cheplygina et al., 2019; Ruiz-Muñoz et al., 2015). 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.

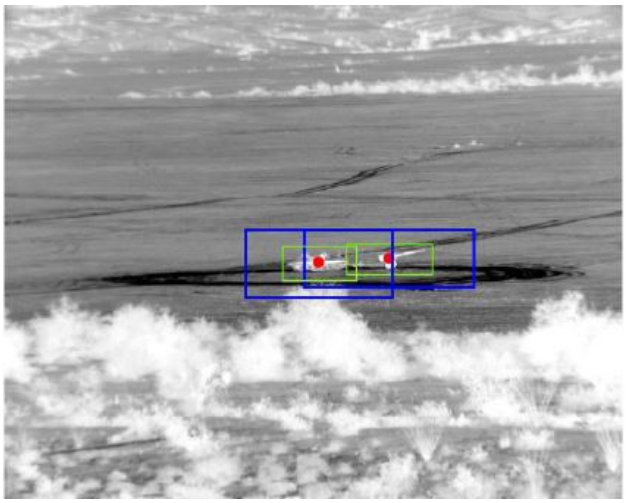


Figure 1-1: A sample frame from the DSIAC MS-003-DB MWIR dataset. Two targets are shown with canonical bounding boxes (green) and relaxed bounding boxes (blue). Red dots represent the centers of the target objects.

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.

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

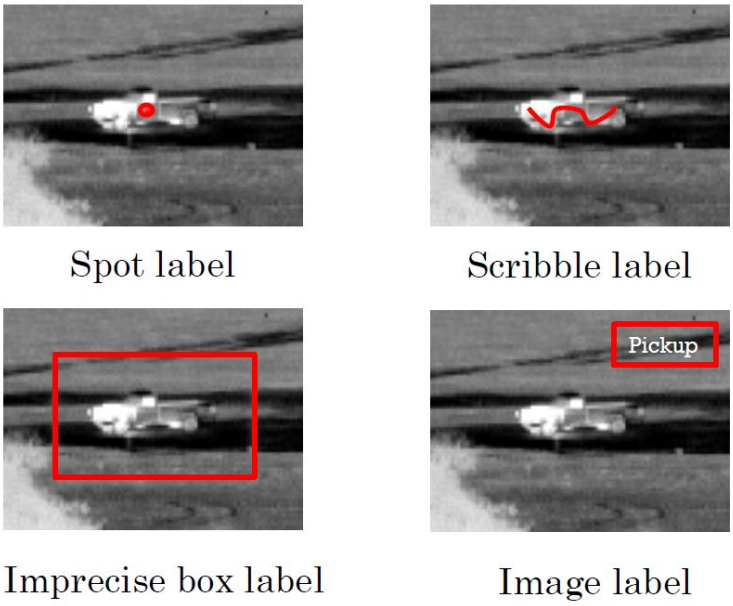


Figure 1-2: Examples of weakly-labeled infrared imagery. The images demonstrate various forms of weak groundtruth around a pickup truck taken with a mid-wave infrared camera. The images show spot, scribble, imprecise bounding box and image-level labels, respectively.

(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.

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. Another way to formulate this problem would be to consider sets of these image patches, where a negative bag would only include samples of background patches, while a positive bag would contain at least one patch that held target or part of a target. 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 generative model which effectively describes the positive and/or negative classes, or training a classifier to discriminate between target and background samples or bags (Ghaffarzadegan, 2018). The success of both learning frameworks, however, is highly dependent on the feature representations of the data and the metrics used to measure dissimilarity. Many feature learning approaches in the literature use supervised learning to obtain discriminative feature representations, or use supervised methods to fine-tune unsupervised feature extraction. *However, this cannot be done directly in MIL because of the uncertainty on the labels (Carbonneau et al., 2016).*

Another approach to obtain useful feature representations is through the application of *manifold learning*, also commonly referred to as *dimensionality reduction* (DR), *feature*

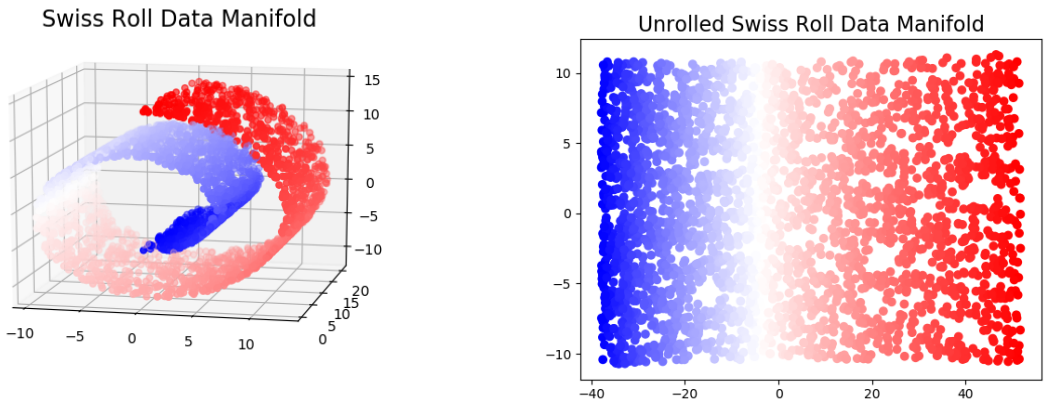
embedding, *geometric machine learning* or *representation learning* in the literature. The goal of manifold learning can be posed as discovering intrinsic (often lower-dimensional) features from the data which meet an overarching objective, such as: preserving variance, finding compressed representations 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 and provide more appropriate similarity metrics 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. As mentioned previously, 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. Additionally, the works in (Ping et al., 2010; Saehoon Kim and Seungjin Choi, 2010; Chai et al., 2014) use adaptations of LDA to project data into a low-dimensional space. All of these methods rely on finding linear projections which will adequately separate positive and negative bags in the embedding space. However, these approaches fail when the the target and background data exhibit highly-curved structure in

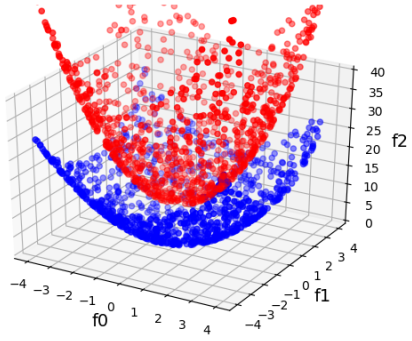


(a) (b)

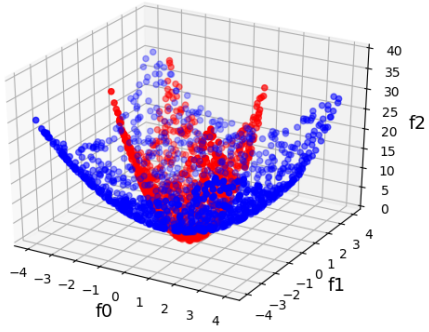
Figure 1-4: (a) This dataset is known as the Swiss Roll and it depicts a 2-dimensional manifold embedded in 3 dimensions. (b) The Swiss Roll unfolded according to the geodesic path around the manifold.

the feature space. For example, consider the data shown in Figure 1-4. This image shows 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 the target class while the blue represent the background class. 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 the manifold according to 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 deemed unnecessary. It would, therefore, be beneficial to develop multiple instance dimensionality reduction approaches which are capable of learning nonlinear manifold structure.

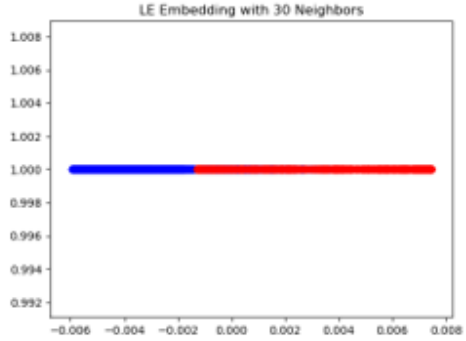
Dimensionality reduction approaches in the literature require instance-level labels or neglect class label information entirely. However, sample-level labels are not available under the MIL framework and unsupervised approaches are typically sub-optimal for discrimination.



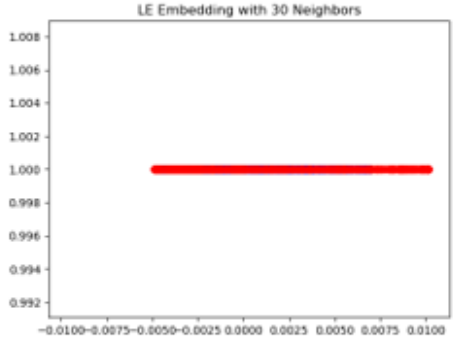
(a) Separable 3-dimensional quadratic surfaces.



(b) Non-separable 3-dimensional quadratic surfaces.



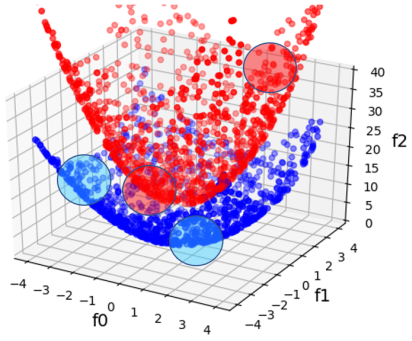
(c) 1-dimensional embedding of separable quadratic surfaces.



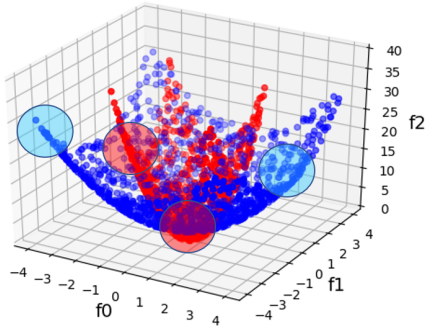
(d) 1-dimensional embedding of non-separable quadratic surfaces.

Figure 1-5: Unsupervised embedding of quadratic surfaces using Laplacian Eigenmaps with a K-nearest neighbors graph.

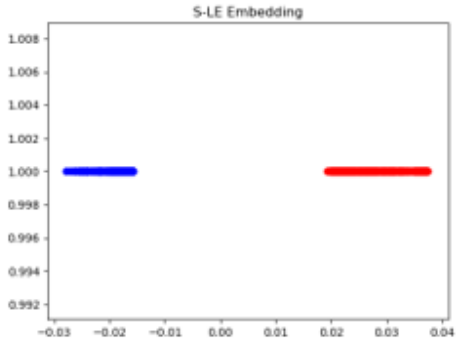
This point is demonstrated by the embeddings of 3-dimensional quadratic surfaces as shown in Figures 1-5 and 1-6. These figures demonstrate a set of separable and non-separable quadratic surfaces, respectively. The use of a traditional, unsupervised dimensionality reduction method to project the data into a 1-dimensional space managed to retain topological ordering of the samples, but failed to promote class discriminability, as shown in Figures 1-5c and 1-5d. As demonstrated in Figures 1-6c and 1-6d, however, a supervised version of the same algorithm learned a mapping which was able to separate the classes in the latent embedding space for both the separable and non-separable quadratic surfaces. This embedding used sample-level labels which, as mentioned, are not available in MIL. Therefore, the objective of the proposed



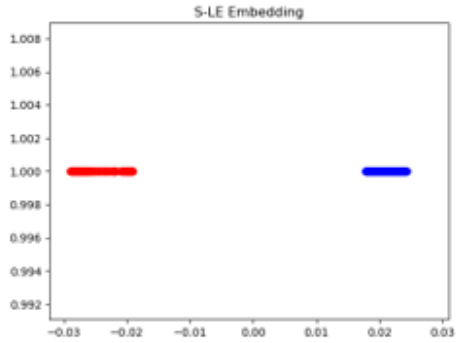
(a) Separable 3-dimensional quadratic surfaces and examples potential bag splits. Blue circles represent negative bags while red circle denote positive bags.



(b) Non-separable 3-dimensional quadratic surfaces and examples potential bag splits. Blue circles represent negative bags while red circle denote positive bags.



(c) 1-dimensional embedding of separable quadratic surfaces using Supervised Laplacian Eigenmaps.



(d) 1-dimensional embedding of non-separable quadratic surfaces using Supervised Laplacian Eigenmaps.

Figure 1-6: Embedding of quadratic surfaces using Supervised Laplacian Eigenmaps.

work is, given only sets of instance and bag-level labels such as those shown in Figures 1-6a and 1-6b, to develop techniques for dimensionality reduction and manifold learning which promote class separation of individual instances in the embedding space. Ideally, instance-level classification accuracy in the embedding space with MIL will match the performance of strictly supervised methods, even with increased label imprecision.

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

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

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 $l_{kn} \in \{-1, +1\}$ 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 perform one of the following tasks: classification, regression, ranking or clustering (Carbonneau et al., 2016).

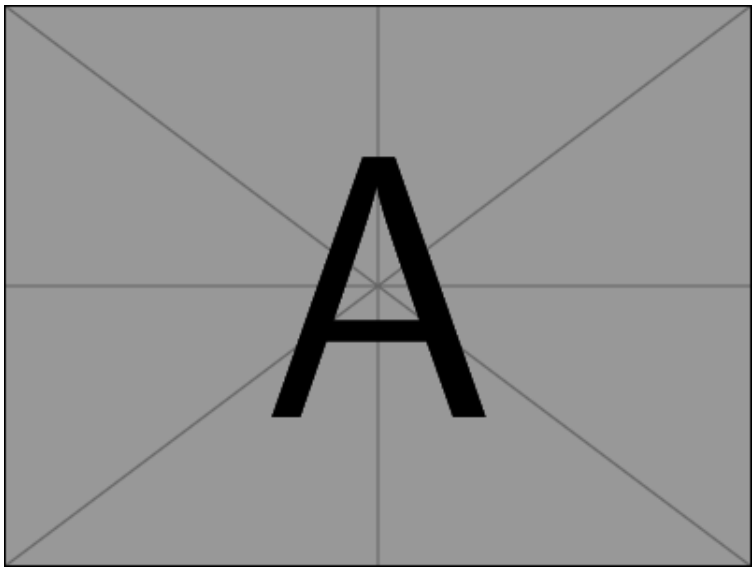


Figure 2-1: Placeholder for examples of positive and negative bag concepts

2.1.1 Multiple Instance Learning with Manifold Bags

It should be noted that while the aforementioned MIL formulation is standard in the literature, Baenko et al. introduced the framework of MIL using manifold bags (Babenko et al., 2011). Their work claimed that instead of finite sets of instances, bags are inherently better represented as low-dimensional manifolds in a high-dimensional feature space. They considered the scenario of classifying whether or not an image contained a face. In this scenario, the entire image was considered to be a bag, and patches of the image were considered as individual instances. It was assumed that the collection of instances collectively formed a low-dimensional manifold. This work laid the groundwork for learning with manifold bags and proved PAC learnability under this framework. While it was worth mentioning this work as it aligns with the objective of manifold learning discussed in the proposed work, the remainder of this literature review focuses solely on the standard MIL formulation.

2.1.2 Tasks

Multiple instance learning in the literature can be broadly categorized into four tasks: classification, regression, ranking and clustering (Carbonneau et al., 2016). MIL classification can be performed at either the bag or instance level. The goal is to assign a class label to either the set of instances or the individual instances themselves. MIL regression consists of assigning a real-valued label to a bag (or instance) instead of a class label. A few methods have been proposed to rank bags or instances instead of assigning a class label or score. This problem differs from regression because the goal is not to obtain an exact real-valued label, but to compare the magnitude of scores to perform sorting. The clustering task consists of finding clusters or structure among a set of unlabeled bags. Alternatively, clustering can be performed within bags in attempt to distinguish between the positive and negative instances. Classification and ranking are the most pertinent tasks to the proposed work, and will thus be discussed in detail.

2.1.3 Multiple Instance Classification

The standard supervised learning task is to learn a classifier based on a training set of feature vectors, where each feature vector is paired with and associated class label. In the *Multiple Instance Classification* (MIC) task, the goal is to learn a classifier based on a training set of bags, where each bag is a set of feature vectors known as instances. In this setting, each bag is paired with an associated binary class label; however, the labels of each instance in the sets are unknown.

2.1.3.1 Space Paradigms

The MIC problem has been formulated under three paradigms: Instance-Space, Bag-Space and Embedded-Space (Amores, 2013). Each paradigm is categorized according to how information presented in the MI data is exploited. In the *Instance-Space* (IS) paradigm, the discriminative information is considered to lie at the instance-level. An instance-level classifier is trained to separate the true positive instances from the true negative instances. Given an instance-level classifier, a bag-level classifier can be developed by simply aggregating the

instance-level scores in a test bag. This paradigm is based on local, instance-level information. In the *Bag-Space* (BS) paradigm, the discriminative information is considered to lie at the bag-level. Under this paradigm, each bag is treated as a whole entity, and the learning process discriminates between entire bags. This paradigm is based on global, bag-level information. Considering that the bag space is a non-vector space, current BS methods make use of non-vectorial learning techniques which define distance metrics as a way to compare bags. In the *Embedded-Space* (ES) paradigm, each bag is mapped to a single feature vector which captures the relevant information about the entire bag. Consequently, the learning problem transforms into a standard supervised problem, where each feature vector is paired with an associated (bag-level) label. Similar to the BS paradigm, the ES paradigm is also based on global, bag-level information. However, the difference between the two paradigms lies in the way bag-level information is extracted. In the BS paradigm, information is extracted implicitly through the definition of the distance or kernel function. Alternatively, information is extracted explicitly in the ES paradigm through the definition of the mapping function from the bag-space to the vector space. Figure 2-2 demonstrates the differences between standard supervised classification and the three MIL classification space paradigms. Apart from space paradigm, MIL classification methods in the literature can be organized according to the their primary approaches toward learning. A review of prominent MIL classification methods in the literature is provided, categorized according to learning approaches.

2.1.3.2 MIL Classification Approaches

MIL classification approaches in the literature can be categorized by the underlying principle used for learning. The categories discussed in this review are: Axis-Parallel Concepts, Maximum Likelihood, Distance-Based, Maximum Margin, Deep Learning, Probabilistic Graphical Methods and Ensembles of Classifiers. Each approach is reviewed in the following.

Learning Axis-Parallel Concepts:. Learning Axis-Parallel Concepts are among the first group of methods used to solve MIL problems (Dietterich et al., 1997). The foundation of this

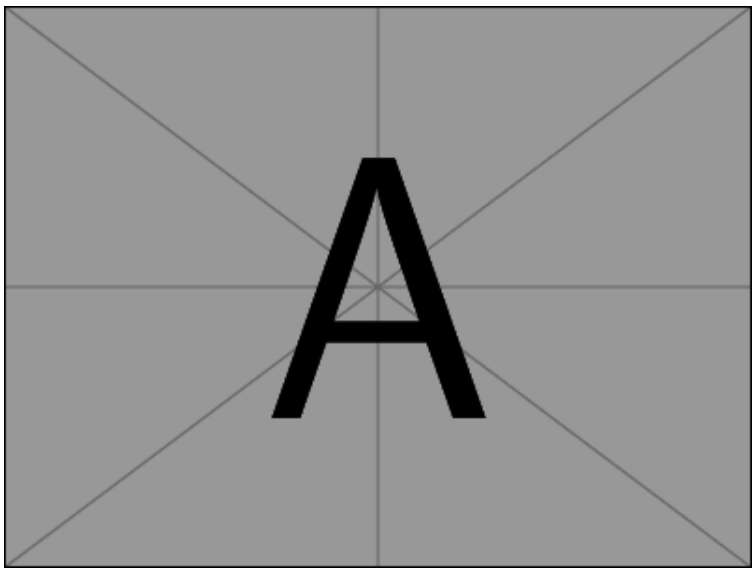


Figure 2-2: MIL classification space paradigm.

category is based on the method of *Axis-Parallel Rectangles* (APR), proposed by Dietterich et al. in the 1990's. An axis-parallel hyper-rectangle is a set of thresholds (one for each feature dimension) that is used to discriminate between two classes. It can also be viewed as an overlap or aggregation region of true positive instances in the feature space. The goal of APR is to find an axis-parallel hyper-rectangle in the feature space to represent the target concept (Ghaffarzadegan, 2018; Bocinsky et al., 2019; Jiao, 2017). A disadvantage of these approaches is that they can fail to neglect a majority of data in a large bag sample.

Maximum Likelihood:. Similar to traditional *Maximum Likelihood Estimation* (MLE), the objective of maximum likelihood in the MIL setting is to train a classifier which maximizes the likelihood of the data. The most prominent of these methods is *Diverse Density* (DD) (Maron and Lozano-Pérez, 1998), which considers each bag as a manifold describing the instances. The goal of learning is to discover a prototype from the training data which maximizes the DD measure. Diverse Density is essentially a measure of the intersection of positive bags minus the union of the negative bags. By maximizing DD, a point of intersection can be found which is close to at least one instance from the positive bags while being as far as possible from all points in the negative bags (Ghaffarzadegan, 2018). Zhang et al. later

proposed *Expectation-Maximization Diverse Density* (EM-DD) (Zhang and Goldman, 2002). EM-DD extends DD by viewing the relationships between instances and their corresponding bag's labels as latent variables. In other words, it asks which instance in the set corresponds to the bag's label (Du, 2017). In the *E-step*, an the instance from each bag which is the most probable for providing the bag its label is selected. Then in the *M-step*, a new concept point is found by maximizing DD with gradient ascent. This process is iterated until a stopping criteria is met, however, it will stop naturally as there only a finite number of instance combination that the algorithm can pick. Multiple instance maximum likelihood approaches have been used in a variety of problems, such as: stock selection, person identification, scene classification, hyperspectral target classification and explosive hazard detection (Maron and Lozano-Pérez, 1998; Maron and Ratan, 1998; Zare et al., 2018, 2015; Bocinsky et al., 2019; McCurley et al., 2019).

Distance-Based:. A simple approach for classification in the supervised learning paradigm is to compare distances of test points to samples in the training set. Gartner et al. defined the MI-Kernel (Gärtner et al., 2002) by regarding each bag as a set of features and applying a set kernel directly to compare similarity. The *Citation-kNN* algorithm (Du, 2017; Carbonneau et al., 2016; Zhou and Zhang, 2003; Saehoon Kim and Seungjin Choi, 2010) is a nearest-neighbor style classifier which borrows the idea of scientific citers and references when considering a bag's label. *References* are simply the nearest neighbors of a bag, while *citers* are the bags which have the query bag in it's C-nearest neighbors. The vanilla citation-kNN uses the *minimal Hausdorff distance* to measure bag similarity. The Hausdorff distance between two bags $\boldsymbol{B}$ and $\boldsymbol{B}'$ is defined as:

$$H(\boldsymbol{B}, \boldsymbol{B}') = \max\{h(\boldsymbol{B}, \boldsymbol{B}'), h(\boldsymbol{B}', \boldsymbol{B})\} \tag{2-2}$$

and

$$h(\boldsymbol{B}, \boldsymbol{B}') = \max_{\boldsymbol{b} \in \boldsymbol{B}} \min_{\boldsymbol{b}' \in \boldsymbol{B}'} ||\boldsymbol{b} - \boldsymbol{b}'|| \tag{2-3}$$

where $\boldsymbol{b}$ and $\boldsymbol{b}'$ are instances in bags $\boldsymbol{B}$ and $\boldsymbol{B}'$, respectively. Intuitively, the minimal Hausdorff distance is the smallest value H such that every instance in $\boldsymbol{B}$ has a point of $\boldsymbol{B}'$ within distance H, and every instance in $\boldsymbol{B}'$ has an instance in $\boldsymbol{B}$ within a distance of H. Variants of citation-kNN include Bayesian citation-kNN and fuzzy citation-kNN (Du, 2017). Additionally, the minimal Hausdorff distance has been substituted for the Earth Mover's Distance (EMD), Chamfer distance and specialized bag-distance kernels (Amores, 2013).

Two alternative distance-based MIL classifiers are MIGraph and miGraph (Zhou et al., 2009). Both methods consider bags as graphs to capture interdependence between instances. The distance measures used to compare bags is what separates the two methods. MIGraph explicitly maps every bag to an undirected graph and uses a graph kernel or metric such as graph edit distance to discriminate between positive and negative bags. Alternatively, miGraph implicitly constructs graphs by defining affinity matrices between instances and uses clique information to help distinguish positive from negative bags.

Maximum Margin:. The concept of weakly-supervised maximum margin learning has been explored under a variety of techniques, the primary being *Support Vector Machines* (SVM) and *Metric Embedding*, which will be discussed in Section 2.4. Andrews et al. proposed two SVM methods under the MIL framework, namely, mi-SVM and MI-SVM, for instance-level and bag-level classification, respectively (Carbonneau et al., 2016; Du, 2017; Jiao, 2017). The goal of a SVM is to train a classifier to maximize the margin between a small subset of training examples (called *support vectors*) and the decision hyper-plane (Murphy, 2012). Both MIL SVM methods follow a similar procedure to EM-DD. In MI-SVM, each positive bag is represented by a single instance which is considered to be the "most positive instance" in the bag. Optimization alternates between learning a decision boundary with SVM and selecting the positive bag representatives given the new classifier. In contrast, mi-SVM considers all points in the positive bags while learning the decision boundary. Every point in the positive bags is provided a positive label. The instance labels are refined iteratively under the constraint of the standard MIL assumption, that at least one instance from each positive bag must lie on

the positive side of the decision hyper-plane (Cao et al., 2016). MissSVM is a semi-supervised max-margin approach which considers the instances in positive bags as unlabeled, and enforces a constraint that at least one of them is positive (Zhou and Xu, 2007). DD-SVM and *Multiple-Instance Learning via Embedded Instance Selection* (MILES) are both embedding-space methods which convert MIL into a standard supervised problem (Yixin Chen et al., 2006). DD-SVM trains an SVM in a feature space constructed from a mapping defined by the local maximizers and minimizers of the DD function. MILES maps each bag into a feature space defined by the instances in the training bags via an instance similarity measure. A 1-norm SVM is applied to simultaneously select the important features and construct classifiers (Ruiz-Muñoz et al., 2015). Additionally, Xiao et al. developed an ensemble method in which the base classifier enforces a margin between optimal hyper-spheres while enclosing at least one instance from each positive bag inside the ball (Xiao et al., 2017).

Neural Networks and Deep Learning:.

Recent developments in deep learning have also made their way into the MIL literature under the assumption that useful features can be learned by the networks using only bag-level labels. Gao et al. used *convolutional neural networks* (CNN) with count-based region selection to perform weakly-supervised object localization (Gao et al., 2017). Ilse et al. modeled the MIL problem as learning the Bernoulli distribution over the bag labels, where the label probability was parameterized by a neural network (Ilse et al., 2018). Ilse's approach employs attention-based MIL pooling as a way to visualize which instances the network selects as being positive. A multi-instance multi-scale CNN to detect regions of interest in medical images (Li et al., 2019) was proposed by Li et al. Li's work introduced "top-k pooling" to aggregate feature maps of varying scales and spatial dimensions, allowing the model to be trained using weak, MIL annotations. Wang et al. explored the use of a U-Net segmentation network architecture to obtain pixel-level ground-cover classification from image-level labels (Wang et al., 2020). A discriminative *variational autoencoder* (VAE) was used by Ghaffarzadegan to maximize the difference between latent representations of positive and negative instances

(Ghaffarzadegan, 2018). Tu et al. developed an end-to-end *graph neural network* which treats each bag as a graph (Tu et al., 2019). Each bag is passed through the network to obtain a feature representation encapsulating the structural information present in the bag. Deep learning MIL approaches have been explored in a wide variety of applications, including: retinal image classification (Tu et al., 2019), histopathology classification (Ilse et al., 2018), object localization (Gao et al., 2017) and region-of-interest proposal in medical images (Li et al., 2019).

Probabilistic Graphical Methods:. *Probabilistic graphical models* (PGMs) are powerful tools used to capture inter-relations between random variables and learn structured models. In some problems, data exhibits an underlying structure between instances or bags that is more complex than simple co-occurrence. Capturing this structure may lead to better classification performance (Carbonneau et al., 2016). Deselaers and Ferrari proposed a multi-instance conditional random field, MI-CRF (Deselaers and Ferrari, 2010). In this method, bags are modeled as nodes in a CRF, where each node can take one of the instances in the bag as its state. Classification corresponds to selecting one instance (positive bag) or selecting no instances (negative bag). Instance selection is formulated as inference in the CRF. This lets all bags to be considered jointly in training and testing. Thus, bags are jointly classified based on unary instance classifiers and pairwise dissimilarity measurements. Hajimirsadeghi and Mori introduced a max-margin classification scheme using Markov networks (Hajimirsadeghi and Mori, 2017). Yuksel et al. developed a multiple instance *Hidden Markov Model* (MI-HMM) for use in landmine detection (Yuksel et al., 2015, 2012). In this scenario, each bag is associated with a label, however, the bags can be composed of time sequences of variable length. A noisy-OR relationship is assumed between the sequences within each bag and the joint probability of the bags of sequences and the corresponding labels for the bags is maximized with a stochastic expectation maximization. PGMs have proven to work well on MIL problems exhibiting time series data and data with structural dependence.

Dictionary Learning:. *Dictionary Learning* is a method of learning how to reconstruct a dataset from a much smaller set of building blocks called *atoms* (Cook, 2015). Given a training data set, the goal is to learn the set of atoms and sparse weights which reconstruct the data. At test, a bag or instance can be classified based on the atoms which effectively reconstruct the sample. Multi-Instance Dictionary Learning (MIDL) uses bag-level information with a least angle regression to alternatively learn a dictionary which represents the training data and regression weights for classificationi. Max-Margin Multiple Instance Dictionary Learning (MMDL) adopts the idea of the bag of words (BoW) model and trains a set of linear SVMs as codebooks (Jiao, 2017). Functions of Multiple Instances (FUMI) is a supervised technique which tries to learn target and background dictionaries such that a target instance can be written as a linear combination of a single target concept and multiple background atoms . This formulation considers target instances that may contain portions of background signature, as with sub-pixel target detection in hyperspectral imagery. A known problem with FUMI is that it does not work well with noisy labels. To account for this problem, extended Functions of Multiple Instances (eFUMI), uses bag-level labels to identify the data (Cook, 2015). A function is built in to determine whether a point labeled as target actually contains a portion of the target dictionary atoms. Task-driven extended Functions of Multiple Instances (TD-eFUMI) adopts the MI aspect of eFUMI and Task-Driven Dictionary learning to simultaneously learn target and background dictionaries in conjunction with a classifier (Cook et al., 2016). Zare et al. proposed the Multiple Instance Adaptive Cosine/Coherence Estimator and Spectral Matched Filter (MI-ACE and MI-SMF) (Zare et al., 2018). These methods learn discriminative prototypes under the multiple instance learning framework to classify instances. However, these methods inherently consider only a single target concept. In order to capture intra-class variation among target instances, Multi-Target MI-ACE/SMF were proposed (Bocinsky, 2019). Finally, Jaio et al. presented Multiple Instance Hybrid Estimator (MI-HE) to learn multiple target and background concepts by maximizing the probability that positive bags are labeled as positive and negative bags are labeled as negative under a noisy-OR model (Jiao

et al., 2018; Bocinsky, 2019). Multiple instance dictionary learning problems have been applied successfully to sub-pixel hyperspectral target detection and landmine detection tasks (Bocinsky, 2019; Zare et al., 2015, 2018; Cook et al., 2016; Jiao et al., 2018).

Ensembles of Classifiers:. *Ensemble learning* paradigms train multiple versions of a base classifier and aggregate the results to achieve a stronger classifier than any of the individuals. Ensemble methods are typically broken into two realms: *bagging* and *boosting*. Bagging employs boostrap sampling to generate several training subsets from the original training set, then trains a learner on each data subset. The predictions from each component learner are aggregated in order to provide a final class score. Zhou et al. studied whether ensemble learning paradigms could be used to enhance MI learners by applying bagging on base MI learners, namely, Diverse Density and Citation K-NN (Zhou and Zhang, 2003). Random Forest classifiers operate under the bagging paradigm and use a technique called *divide-and-conquer*. These classifiers deploy an ensemble of decision trees which iteratively divide the feature space and make simple thresholding decisions. The predicted class labels provided by each tree in the forest are combined to give a final class label. Leistner et al. proposed MIForest which combines the random forest learning algorithm with MIL (Leistner et al., 2010). Since only bag-level labels are known, MIForest treats the label of each instance as a random variable defined over a space of probability distributions. The instance labels are disambiguated by iteratively searching for distributions which minimize the overall classification error.

The other paradigm of ensemble learning is called boosting. The goal of boosting is, using the entire training set, to find a weighted combination of weak learners (that may perform only slightly better than chance), such that the combination produces a strong classifier with high classification accuracy (Zhang et al., 2006). Several MI boosting approaches have been proposed in the literature. Section 2.1.4 discusses a popular variant, MIL-Boost, in detail.

2.1.4 Multiple Instance Boosting (MIL-Boost)

Gradient Boosting Overview. In the standard supervised learning setting, the goal of binary classification is to learn a classification function $h : \mathcal{X} \to \mathcal{L}$ which maps data in $\mathcal{X}$ to a

binary class label $\mathcal{L} \in \{-1, +1\}$. The objective of boosting is to train a classifier of the form

$$\boldsymbol{h}(\boldsymbol{x}) = \sum_{t=1}^{T} \alpha_t h_t(\boldsymbol{x}) \tag{2-4}$$

where each $h_t : \mathcal{X} \rightarrow \mathcal{L}$ is a *weak learner* whose performance may only be slightly above chance, and the weights α_t are each weak learners' relative importance (Babenko et al., 2008). *Boosting* combines multiple weak learners into a single *strong* classifier with low classification error. This is also known in the literature as *ensembling*. In each phase of training, samples classified incorrectly are given more weight in order to improve classification performance in the next iteration. The response of each weak classifier h_t is given as the maximum response over all instances in the training set:

$$h_t = \arg\max_{h} \sum_{n=1}^{N} w_n h(\boldsymbol{x}_n) \tag{2-5}$$

MIL-Boost. Boosting under the MIL framework was originally proposed by Zhang et al. and is known as *MIL-Boost* (Zhang et al., 2006). Under MIL, it is assumed that every instance has a true label $l_{kn} \in \{-1, +1\}$. A bag is labeled positive if at least one of its instances are positive, and a bag is labeled negative if every instance is negative. This means that the label of a bag is provided as the max label over all instances in the bag:

$$L_k = \max_{n}(l_{kn}) \tag{2-6}$$

In this setting, the goal is to learn a classifier $\boldsymbol{h}$ using only bag-level labels such that $\max_n(\boldsymbol{h}(\boldsymbol{x}_{kn})) = L_k$. The score given to a sample is $l_{kn} = \boldsymbol{h}(\boldsymbol{x}_{kn})$ and $\boldsymbol{h}(\boldsymbol{x}_{kn}) = \sum_{t=1}^{T} \alpha_t h_t(\boldsymbol{x}_{kn})$ which is a weighted sum of predicted labels from each of the weak classifiers. Obviously, the sign of the prediction from the strong classifier provides the class label for the instance.

A natural objective is to minimize the negative log-likelihood between instances and their predicted labels. This can also be done at the bag-level. The probability that an instance is

positive is given by the standard logistic function

$$p_{kn} = \frac{1}{1 + \exp(-l_{kn})} \tag{2-7}$$

and the probability that bag k is positive is given by a "noisy OR", $p_k = 1 - \prod_{n \in k}(1 - p_{kn})$. Therefore, the likelihood assigned to a set of training bags is given by:

$$\mathcal{L}(\boldsymbol{h}) = \prod_{k=1}^{K} p_k^{L_k}(1 - p_k)^{(1-L_k)} \tag{2-8}$$

where $L_k \in \{0, 1\}$ is the actual label of the bag.

Following the idea of gradient boosting, the weight on each training instance is given as the derivative of the log-likelihood with respect to a change on the score given to the instance. Thus gradient descent can be used to update the the strong classifier. Pseudo-code for MIL-Boost is provided in Algorithm 1.

Algorithm 1 MIL-Boost

Input: Dataset $\{\boldsymbol{B}_1, \ldots, \boldsymbol{B}_K\}$, $\{L_1, \ldots, L_K\}$, $L_k \in \{-1, +1\}$

1: **for** $t = 1$ to T **do**
2: Compute weights $w_{kn} = -\frac{\partial \mathcal{L}}{\partial \boldsymbol{h}_{kn}}$
3: Train weak classifier h_t using weights $|w_{kn}|$
$h_t = \arg\min_h \sum_{kn} \mathbf{1}(h(\boldsymbol{x}_{kn} \neq L_k))|w_{kn}|$
4: Find α_t via line search to minimize $\mathcal{L}(\boldsymbol{h})$
$\alpha_t = \arg\min_\alpha \mathcal{L}(\boldsymbol{h} + \alpha h_t)$
5: Update strong classifier $\boldsymbol{h} \leftarrow \boldsymbol{h} + \alpha_t h_t$
6: **end for**

Each round of boosting consists of updating the weak learners according to the instances they misclassified and updating the strong classifier according to the new weights calculated over the weak learners. The goal of MIL-Boost is to be able to correctly classify each instance in a test bag as being positive or negative such that the label of the bag is given as the maximum label over all instances in the bag.

2.1.5 Multiple Instance Ranking

Another primary task in MIL is *ranking* (Carbonneau et al., 2016). The ranking problem is different from classification because, instead of providing a binary class label, the objective

is to order a set of bags (Bergeron et al., 2008; Bergeron et al., 2012) or instances (Hu et al., 2008) according to preference for a particular task. Ranking is a key task under *Preference Learning*, or learning to predict preferences on a set of alternatives, which are often represented in the form of an order relation (Fürnkranz and Hüllermeier, 2003). The statement that $\boldsymbol{x}$ is preferred to $\boldsymbol{x}'$ can be simply expressed as an inequality relation $f(\boldsymbol{x}) > f(\boldsymbol{x}')$, where $\boldsymbol{x}$ and $\boldsymbol{x}'$ are instances, and f defines a preference function. For example, given a news story about the Olympics, one might prefer to give it the label "sports" rather than "politics" or "weather". Alternatively, one might prefer to label one bag or instance as "positive" over another. In machine learning, the preference learning problem is often analyzed in two cases: *learning instance preference* and *learning label preference* (Chu and Ghahramani, 2005). Under the scenario of learning instance preferences, the training set consists of a set of pairwise preferences between instances. The objective is to learn the underlying ordering from the set of pairwise distances (such as ordering bags from the "most positive" to "least positive"). Alternatively, the goal of label preference learning is to order a pre-defined set of labels for each individual instance (such as ordering the preference of "positive" or "negative" on each individual bag or instance) (Dekel et al., 2004; Aiolli and Sperduti, 2004).

Ranking under the multiple instance framework was proposed in (Bergeron et al., 2008) for predicting hydrogen atom grouping in computational chemistry. The method proposed by Bergeron et al. is called MIRank (Bergeron et al., 2012). MI ranking differs from MIC in that the label for each bag is not known. Instead, the MI ranking algorithms are provided preference information between pairs of bags. MIRank considers the partial ranking problem, which inherently exhibits three levels of structure: items (instances) belong to bags and bags belong to boxes. The objective is to learn a ranking function that can identify the preferred bag in each box. A concrete example where this framework can be applied is in learning positive target concepts. For a set of video frames (boxes), one may want to predict the smaller image chips (bags) that have the highest probability of containing a target object (positive instances). MIRank uses a linear prediction function to rank instances in individual bags. The ranking of

instances $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in bags I and J is guided by the preference information between I and J. The work in (Hu et al., 2008) introduced *multiple-instance ranking* based on a max margin framework. In this setting, images were represented by sets of regions and the goal was to rank images according to relevance to a keyword. Assuming the preference relationship that $\boldsymbol{x}_m$ is preferable to $\boldsymbol{x}_n$ is denoted by $\boldsymbol{x}_m \succ \boldsymbol{x}_n$, the goal is to induce a ranking function $f : \mathbb{R}^D \rightarrow \mathbb{R}$ that fulfills the set of constraints

$$\forall \boldsymbol{x}_m \succ \boldsymbol{x}_n : f(\boldsymbol{x}_m) > f(\boldsymbol{x}_n) \tag{2-9}$$

The value of $f(\boldsymbol{x}_n)$ is referred to as the ranking score of $\boldsymbol{x}_n$ and is typically a linear function $f(\boldsymbol{x}_n) = \langle \boldsymbol{w}, \boldsymbol{x}_n \rangle = \boldsymbol{w}^T \boldsymbol{x}_n$. Adding slack variable ζ_{mn}, the optimization problem can be solved with with the following objective:

$$\begin{aligned} \min_{\boldsymbol{w},\boldsymbol{\zeta}} \quad & \frac{1}{2}||\boldsymbol{w}||^2 + \gamma \sum_{m,n} \zeta_{mn} \\ \text{s.t.} \quad & \forall \boldsymbol{x}_m \succ \boldsymbol{x}_n : \langle \boldsymbol{w}, \boldsymbol{x}_m \rangle \geq \langle \boldsymbol{w}, \boldsymbol{x}_n \rangle + 1 - \zeta_{mn} \\ & \forall m, n : \zeta_{mn} \geq 0 \end{aligned} \tag{2-10}$$

which is referred to as a *ranking SVM*. Assuming the optimal solution is $\boldsymbol{w}^*$, the ranking score of a test point $\boldsymbol{x}'$ is given as $f(\boldsymbol{x}') = \langle \boldsymbol{w}^*, \boldsymbol{x}' \rangle$. This framework assumes each sample $\boldsymbol{x}_n$ is a single instance. In multiple-instance ranking, we are given a set of preference relations between bag pairs and it is assumed that the score of a bag $\boldsymbol{B}_k$ is determined by the scores of the instances it contains

$$h(\boldsymbol{B}_k) = h\left(\{f(\boldsymbol{x}_n)\}_{n=1}^{N_k}\right) \tag{2-11}$$

Under this formulation, the objective can be re-written to consider bag scores as

$$\begin{aligned} \min_{f \in \mathcal{H}, \boldsymbol{\zeta}} \quad & \frac{1}{2}||f||_{\mathcal{H}}^2 + \gamma \sum_{m,n} \zeta_{mn} \\ \text{s.t.} \quad & \forall \boldsymbol{B}_m \succ \boldsymbol{B}_n : h(\boldsymbol{B}_m) \geq h(\boldsymbol{B}_n) + 1 - \zeta_{mn} \\ & \forall m, n : \zeta_{mn} \geq 0 \end{aligned} \tag{2-12}$$

A bag's score is determined by the scores of its instances. Different functions for providing a bag score have been investigated, including using the max of instance scores, the mean of instance scores and the softmax of instance scores.

Besides MIRank and multiple-instance ranking, Asif et al. recently proposed pyLEMMINGS, which implements locally linear MI ranking by learning a large margin discriminant function from bags with corresponding integer rankings (Asif et al., 2017). While ranking has been successfully applied to text information retrieval, image retrieval (Hu et al., 2008) and bioinformatics (Asif et al., 2017), ranking under the MIL framework is still a relatively unexplored area of research.

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 2-3. 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.

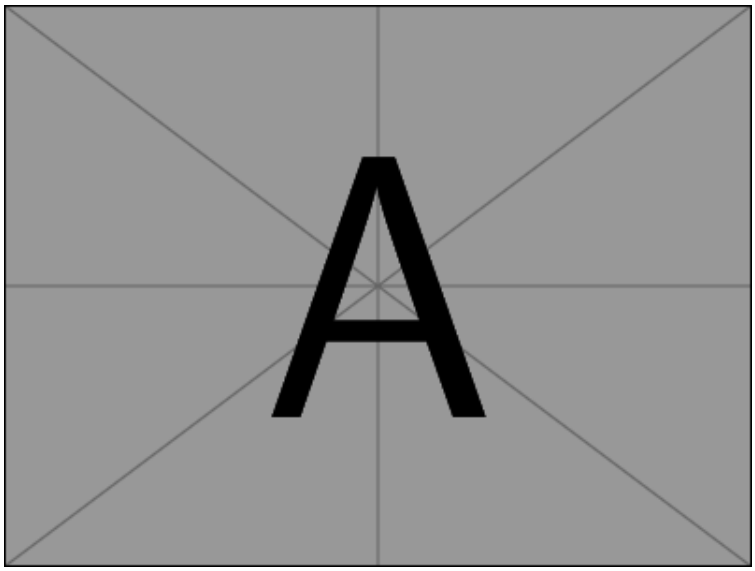


Figure 2-3: Placeholder for example of high D data lying on a low-dimensional sub-manifold.

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 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 $\boldsymbol{X} = [\boldsymbol{x}_1^T, \ldots, \boldsymbol{x}_N^T]$ and $\boldsymbol{Z} = [\boldsymbol{z}_1^T, \ldots, \boldsymbol{z}_N^T]$ are of size $N \times D$ and $N \times d$, 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 brevity (Murphy, 2012). 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 (Kegl et al., 2008). 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)

$$J(\boldsymbol{W}, \boldsymbol{Z}) = \frac{1}{N} \sum_{n=1}^{N} ||\boldsymbol{x}_n - \hat{\boldsymbol{x}}_n||^2 \tag{2-13}$$

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

$$J(\boldsymbol{W}, \boldsymbol{Z}) = ||\boldsymbol{X} - \boldsymbol{W}\boldsymbol{Z}||_F^2 \tag{2-14}$$

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

$$||\boldsymbol{A}||_F = \sqrt{\sum_{m=1}^{M} \sum_{n=1}^{N} a_{mn}^2} = \sqrt{tr(\boldsymbol{A}^T \boldsymbol{A})} = ||\boldsymbol{A}(:)||_2 \tag{2-15}$$

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 2-4 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).

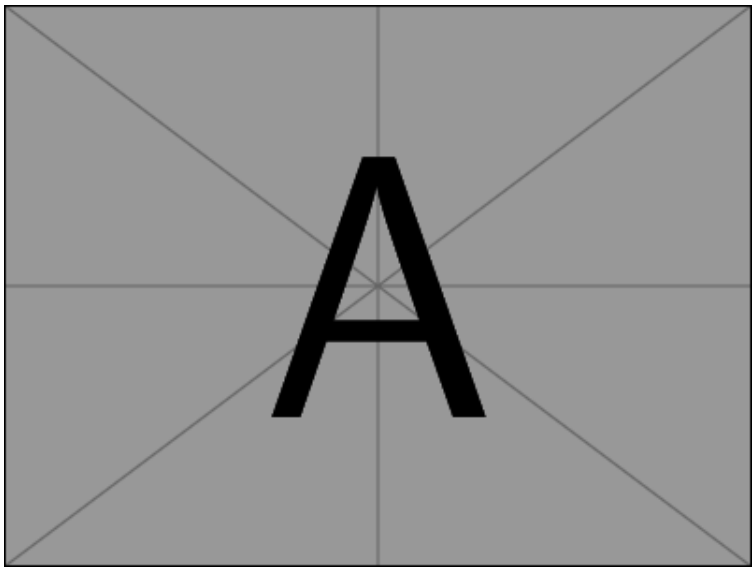


Figure 2-4: Placeholder for PCA projection example.

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?**.

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.

2.2.4.2 Multi-Dimensional Scaling (MDS)

Unsupervised MDS. Most modern manifold learners have theoretical and algorithmic roots in one of three basic dimensionality reduction techniques: PCA, K-means and *Multidimensional Scaling* (MDS). Whereas PCA looks for linear projection bases which are constructed from the eigenvectors of a data covariance or scatter matrix, MDS tries to find a linear projection that preserves pairwise distances as well as possible. This idea is demonstrated by Figure 2-5, where the pairwise distances between samples in the 3-dimensional space are preserved in the 2-dimensional embedding space. While MDS does not construct an embedded manifold explicitly, it holds the status of being the grandfather of "one-shot" (non-iteratve) manifold learners, such as Isomap and Locally Linear Embedding (LLE), which are discussed later in this literature review (Kegl et al., 2008).

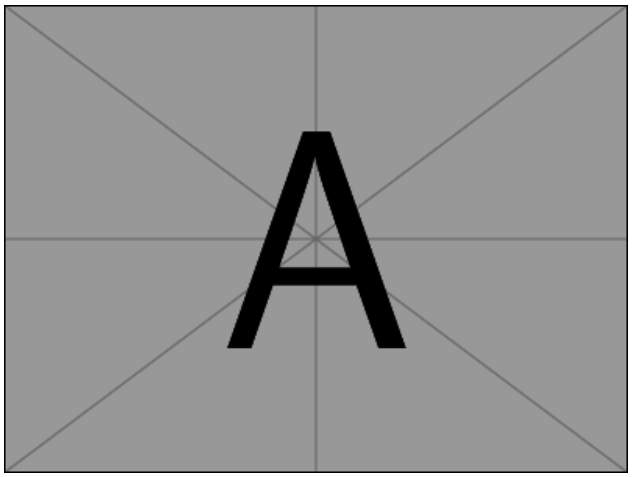


Figure 2-5: Example of MDS distance preservation.

The steps of MDS correspond exactly to those of PCA except that, instead of a scatter matrix $\boldsymbol{S} = \frac{1}{N}\boldsymbol{X}\boldsymbol{X}^T$, MDS operates with a positive semi-definite, dissimilarity matrix $\boldsymbol{D} \in \mathbb{R}^{N \times N}$, where N is the number of data samples and a real, symmetric Gram matrix

$\boldsymbol{K} = \boldsymbol{X}^T\boldsymbol{X}$ (inner-product matrix) where $\boldsymbol{K}_{mn}$ is the inner product between $\boldsymbol{x}_m$ and $\boldsymbol{x}_n$. The problem of MDS is posed as finding d-dimensional Euclidean coordinates for each sample $\boldsymbol{x}_n$ in dataset $\boldsymbol{X}$ such that the Euclidean distances in the low-dimensional embedding space are proportional to the pairwise distances in the input space (Thorstensen, 2009; Sorzano et al., 2014). While the literature poses several cost functions for this task, this review focuses on classical MDS, which is described as follows:

First, the pairwise distance matrix $\boldsymbol{D}$ is computed such that

$$\boldsymbol{D}_{mn} = \mathcal{D}_{\mathcal{X}}(\boldsymbol{x}_m, \boldsymbol{x}_n) = ||\boldsymbol{x}_m - \boldsymbol{x}_n||^2 = (\boldsymbol{x}_m - \boldsymbol{x}_n)^T(\boldsymbol{x}_m - \boldsymbol{x}_n) \tag{2-16}$$

where $\mathcal{D}_{\mathcal{X}}(\cdot,\cdot)$ is a chosen dissimilarity metric (Euclidean distance for classical MDS). Then, the double-centered Gram matrix $\boldsymbol{K}$ is computed by

$$\boldsymbol{K} = -\frac{1}{2}\boldsymbol{H}\boldsymbol{D}\boldsymbol{H} \tag{2-17}$$

where

$$\boldsymbol{H} = \mathbb{I}_N - \frac{1}{N}\boldsymbol{e}\boldsymbol{e}^T \tag{2-18}$$

with $\mathbb{I}_N$ denoting the $N \times N$ identity matrix and $\boldsymbol{e} = (1, \ldots, 1)^T$ the $N \times 1$ column vector of all ones. Multiplying $\boldsymbol{D}$ on both sides by $\boldsymbol{H}$ performs *double centering*, which subtracts the row and column means from $\boldsymbol{D}$ (and adds back the global mean which gets subtracted twice), so that both the row and column means of $\boldsymbol{K}$ are equal to zero. The objective is to find $\boldsymbol{Z} = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N\} \in \mathbb{R}^d$ which minimizes the objective

$$J(\boldsymbol{D}_{\boldsymbol{X}}, \boldsymbol{D}_{\boldsymbol{Z}}) = ||\boldsymbol{K}_{\boldsymbol{X}} - \boldsymbol{D}_{\boldsymbol{Z}}||^2 = \left|\left|-\frac{1}{2}\boldsymbol{H}(\boldsymbol{D}_{\boldsymbol{X}} - \boldsymbol{D}_{\boldsymbol{Z}})\boldsymbol{H}\right|\right|^2 \tag{2-19}$$

Similarly to PCA, this can be solved by a generalized eigenvalue problem

$$\boldsymbol{K}v = \lambda v \tag{2-20}$$

such that

$$\boldsymbol{Z} = \boldsymbol{V}\boldsymbol{\Lambda}^{\frac{1}{2}} \tag{2-21}$$

with $\boldsymbol{\Lambda}^{\frac{1}{2}} = diag(\sqrt{\lambda_1}, \ldots, \sqrt{\lambda_N})$ being a diagonal matrix with entries equal to the square roots of the eigenvalues of $\boldsymbol{K}$ sorted from largest to smallest ($\lambda_1 \geq \lambda_2 \geq \cdots \geq 0$), and $\boldsymbol{V} = \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_N\}$ the corresponding eigenvectors. The low-dimensional embedding coordinates $\boldsymbol{Z} \in \mathbb{R}^{N \times d}$ are obtained by $\boldsymbol{Z} = \{\sqrt{\lambda_1}\boldsymbol{v_1}, \ldots, \sqrt{\lambda_d}\boldsymbol{v_d}\}$ (Chao et al., 2019).

It has been proven that the eigenvalues of Gram matrix $\boldsymbol{K}$ and covariance $\boldsymbol{S}$ are the same, and that the space spanned by MDS and PCA are the identical for any $d \leq rank(\boldsymbol{K}) = rank(\boldsymbol{S})$. This implies that a rotation matrix $\boldsymbol{A}$ could be found such that $\boldsymbol{A}^T\boldsymbol{Z}_{MDS} = \boldsymbol{Z}_{PCA}$ (Sorzano et al., 2014). Additionally, MDS can be computed even if the data observation matrix $\boldsymbol{X}$ is unknown. All that is needed is the Gram matrix or a dissimilarity matrix. This feature potentially allows MDS to be applied in a variety of data-sensitive and privacy-concerned scenarios.

A pitfall of MDS is that it focuses on retaining global pairwise distances as opposed to local distances, which are typically much more important for capturing the geometry of the data (van der Maaten et al., 2007). Several MDS variants have been proposed to address this weakness. A popular variant is known as *Sammon Mapping* and is discussed in Section 2.2.5.9.

Supervised MDS. As with most manifold learning methods in the literature, MDS does not inherently consider class information when learning the embedding function. In attempt to promote class separability in the low-dimensional embedding space, Witten et al. proposed a *Supervised Multidimensional Scaling* (SMDS) (Witten and Tibshirani, 2011). this method follows the idea of traditional MDS where the goal is to find low-dimensional coordinate or *configuration points* $\boldsymbol{z}_n \in \mathbb{R}^d$, such that pairwise distances in the input feature space are preserved in the embedding space. Incorporating class label information, the goal of SMDS is to not only preserve distances, but ensure the coordinate values $z_{mk} > z_{nk}$ when $l_m > l_n, \quad \forall k = 1, \ldots, d$, where l are the instance-level labels and d is the dimensionality

of the embedding space. Considering the binary target classification case, SMDS can be formulated as

$$\min_{\boldsymbol{Z}} \quad \frac{1}{2}(1-\alpha)\sum_{m=1}^{N}\sum_{n=1}^{N}(\boldsymbol{D}_{mn} - ||\boldsymbol{z}_m - \boldsymbol{z}_n||^2) + \alpha \sum_{m:l_m=1}\sum_{n:l_n=2}\sum_{k=1}^{d}\left(\frac{\boldsymbol{D_{mn}}}{\sqrt{d}} - (z_{nk} - z_{mk})^2\right) \tag{2-22}$$

This objective has two terms. The first is the traditional metric MDS *stress*. This term attempts to ensure that the Euclidean distances of two points in the embedding space is the same as the dissimilarity between the points in the input feature space. The second term is the supervised term which enforces that each dimension of the embedded configuration points be larger if belonging to the class with the larger label, and smaller if belonging to the class with a smaller-valued label. The term $\alpha \in [0, 1]$ is a tuning parameter. When $\alpha = 0$, the objective reduces to the MDS stress function. As α increases, however, the objective becomes increasingly more supervised, focused on ensuring class separation of the training data.

A least square regression was applied to estimate the embedding function for out-of-sample test points. SMDS was successfully applied to tasks in data visualization, bipartite ranking and classification of prostate data and USPS handwritten digits.

2.2.4.3 Non-negative Matrix Factorization (NMF)

2.2.4.4 Fisher's Linear Discriminant Analysis (LDA)

Classical LDA. *Linear Discriminant Analysis* (LDA) is a popular method for supervised, linear dimensionality reduction. LDA currently forms the basis for Multiple Instance Learning dimensionality reduction methods exhibited in the literature (Sun et al., 2010; Chai et al., 2014; Zhu et al., 2018; Xu et al.). Whereas PCA tries to project data into a space which maximizes variance, LDA considers class label information and tries to find a transformation which both maximizes between-class (inter-class) dissimilarity and minimizes between-class (intra-class) compactness (Yan et al., 2007; Chao et al., 2019; Sun et al., 2010; Murphy, 2012). This is done by maximizing the ratio between the inter-class $\boldsymbol{S}_b$ and intra-class $\boldsymbol{S}_w$

scatter matrices, defined as:

$$\boldsymbol{S}_w = \sum_{k=1}^{K} \boldsymbol{S}_k \tag{2-23}$$

$$\boldsymbol{S}_k = \sum_{n \in C_k} (\boldsymbol{x}_n - \hat{\boldsymbol{\mu}}_k)(\boldsymbol{x}_n - \hat{\boldsymbol{\mu}}_k)^T \tag{2-24}$$

$$\boldsymbol{S}_b = \sum_{k=1}^{K} N_k (\hat{\boldsymbol{\mu}}_k - \hat{\boldsymbol{\mu}})(\hat{\boldsymbol{\mu}}_k - \hat{\boldsymbol{\mu}})^T \tag{2-25}$$

Here, $\boldsymbol{S}_w$ is the global within-class scatter matrix which is defined as the sum over each individual class' scatter matrix $\boldsymbol{S}_k$, and $\boldsymbol{S}_k$ is essentially an outer product between all samples belonging to class C_k after subtracting the respective empirical class mean $\hat{\boldsymbol{\mu}}_k$. This scatter matrix would be the class covariance if it was normalized by the number of samples N_k in class C_k. However, this normalization constant does not affect the final solution and can thus be ignored. The between-class scatter $\boldsymbol{S}_b$ is defined by the sum of outer products of the differences between the empirical class means $\hat{\boldsymbol{\mu}}_k$ and the global data mean $\hat{\boldsymbol{\mu}}$, weighted by the number of samples in each class. The objective of LDA is then to solve for $\boldsymbol{W}^*$ which maximizes the ratio $J(\boldsymbol{W})$:

$$\boldsymbol{W}^* = \arg\min_{\boldsymbol{W}} J(\boldsymbol{W}) = \arg\max_{\boldsymbol{W}} \frac{|\boldsymbol{W}^T \boldsymbol{S}_b \boldsymbol{W}|}{|\boldsymbol{W}^T \boldsymbol{S}_w \boldsymbol{W}|} \tag{2-26}$$

It has been shown that the optimal projection matrix $\boldsymbol{W}^*$ is the one whose columns are the eigenvectors corresponding to the largest eigenvalues of the generalized eigenvalue problem

$$\boldsymbol{S}_b \boldsymbol{w} = \lambda \boldsymbol{S}_w \boldsymbol{w} \Rightarrow \boldsymbol{S}_w^{-1} \boldsymbol{S}_b \boldsymbol{w} = \lambda \boldsymbol{w} \tag{2-27}$$

Since $\boldsymbol{S}_b$ is the sum of K matrices of rank ≤ 1, this implies that $\boldsymbol{S}_b$ will be of rank $(K-1)$ or less and only $(K-1)$ of the eigenvalues λ will be non-zero. A low-dimensional coordinate representation $\boldsymbol{z}_n \in \mathbb{R}^{(K-1)}$ of sample $\boldsymbol{x}_n \in \mathbb{R}^D$ is given by the linear projection of $\boldsymbol{x}_n$ onto the hyper-plane parameterized by $\boldsymbol{W}^*$, $\boldsymbol{z}_n = \boldsymbol{W}^{*T} \boldsymbol{x}_n$ It should be noted that for LDA, the

dimensionality of the latent space is not a free-parameter, but is always fixed at $d = (K - 1)$, or one less than the number of classes present in the dataset. Equivalently, LDA can be derived by maximum likelihood for normal class-conditional densities where the covariances for each class are assumed to be equivalent (Murphy, 2012). For the special case of binary target classification, the LDA transformation will place every sample onto a single line in 1-dimension, and thus the LDA solution can be simplified:

$$\boldsymbol{w}^* = \arg\max_{\boldsymbol{w}} \frac{\boldsymbol{w}^T \boldsymbol{S}_b \boldsymbol{w}}{\boldsymbol{w}^T \boldsymbol{S}_w \boldsymbol{w}} = \boldsymbol{S}_w^{-1}(\hat{\boldsymbol{\mu}}_2 - \hat{\boldsymbol{\mu}}_1) \tag{2-28}$$

where $\hat{\boldsymbol{\mu}}_1$ and $\hat{\boldsymbol{\mu}}_2$ are the empirical means for classes 1 and 2, respectively. Figure 2-6 demonstrates the differences between PCA and LDA. While PCA projects data onto the axes exhibiting the maximal variation, LDA projects the data into a space which attempts to simultaneously enforce between-class separation and within-class compactness.

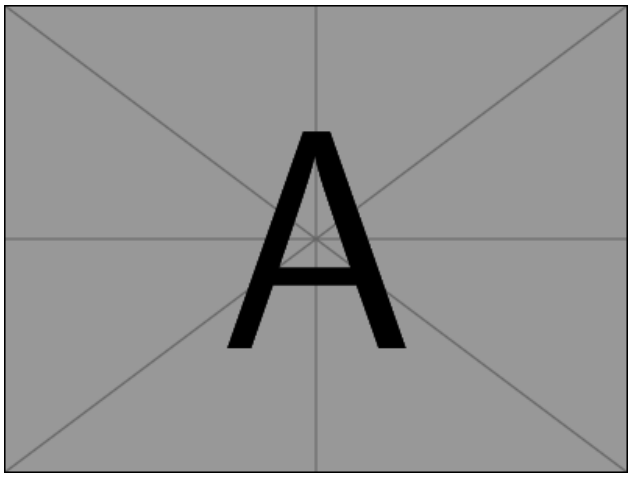


Figure 2-6: LDA example.

Although LDA is the basis for large number of discriminative dimensionality reduction approaches, it does not guarantee class separation in the embedding space. For example, LDA projects data into a space of at most $(K - 1)$ dimensions, however, more features may be necessary for adequate class discrimination. Additionally, LDA is a parametric method which assumes unimodal Gaussian likelihoods. This implies that it may not be able to preserve

complex data structure. Finally, LDA will fail if the discriminatory information is contained in the variance of the data instead of the mean. Despite these pitfalls, LDA has been successfully applied to object detection and recognition tasks (Wang et al., 2016) Many variations of LDA have been developed, such as Non-parametric LDA (Fukunaga and Mantock, 1983), Orthonormal LDA (Wang et al., 2016), Generalized LDA (Baudat and Anouar, 2000) and Multilayer Perceptrons (Webb and Lowe, 1990). Additionally, LDA serves as the foundation for all of the Multiple Instance Learning dimensionality reduction approaches in the current literature (Sun et al., 2010; Chai et al., 2014; Zhu et al., 2018).

2.2.4.5 Locality Preserving Projection (LPP)

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 (Kegl et al., 2008). 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 $\mathbb{R}^D$.

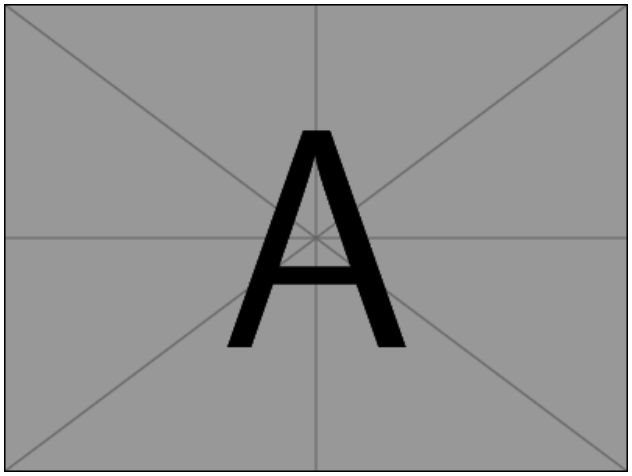


Figure 2-7: Example of a nonlinear manifold.

2.2.5.1 Kernelization

Although each of the manifold learning techniques previously discussed are inherently linear, nonlinear adaptations have been made. One easily extendable approach is to utilize kernel functions as means to provide nonlinearity in the embeddings.

Kernels. A *kernel function* is a real-valued function of two arguments $\kappa(\boldsymbol{x}, \boldsymbol{x}') \in \mathbb{R}$, for $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}$, which maps vectors from the input feature space to a single value in $\mathbb{R}$. The function is typically symmetric (i.e. $\kappa(\boldsymbol{x}, \boldsymbol{x}') = \kappa(\boldsymbol{x}', \boldsymbol{x})$) and non-negative (i.e. $\kappa(\boldsymbol{x}, \boldsymbol{x}') \geq 0$), which implies that it can be interpreted as a measure of similarity (Murphy, 2012). The notion of kernels is very useful in certain applications where data representation is not straightforward, such as representing text documents or molecular structures which can have variable length. Additionally, this allows algorithms to operate directly on the kernel representations. This is useful in data-sensitive scenarios where direct access to the data may not be available.

A popular choice of kernel in manifold learning is the *radial basis function* (RBF) kernel, defined as:

$$\kappa(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{||\boldsymbol{x} - \boldsymbol{x}'||^2}{\beta}\right) \tag{2-29}$$

where β is the bandwidth of the isotropic function. Another popular kernel for text classification is *cosine similarity*, defined by:

$$\kappa(\boldsymbol{x}, \boldsymbol{x}') = \frac{\boldsymbol{x}^T \boldsymbol{x}'}{||\boldsymbol{x}||_2 ||\boldsymbol{x}'||_2} \tag{2-30}$$

This kernel measures the cosine of the angle between vectors $\boldsymbol{x}$ and $\boldsymbol{x}'$ after scaling them onto the unit hyper-sphere. If $\boldsymbol{x}$ and $\boldsymbol{x}'$ are strictly positive vectors (counts in the bag-of-words model, for example), then the kernel provides values in $[0, 1]$, where a value of 0 means the feature vectors are orthogonal and, therefore, have no features in common, and a value of 1 means the vectors are the same.

Some of the nonlinear manifold learning methods in the literature require the kernel function to satisfy the requirement that the *Gram matrix*

$$\boldsymbol{K} = \begin{bmatrix} \kappa(\boldsymbol{x}_1, \boldsymbol{x}_1) & \dots & \kappa(\boldsymbol{x}_1, \boldsymbol{x}_N) \\ & \vdots & \\ \kappa(\boldsymbol{x}_N, \boldsymbol{x}_1) & \dots & \kappa(\boldsymbol{x}_N, \boldsymbol{x}_N) \end{bmatrix} \tag{2-31}$$

be positive definite for any set of inputs $\{\boldsymbol{x}_n\}_{n=1}^N$. This type of kernel is called a *Mercer kernel* or *positive definite kernel*. The importance of the Mercer Kernel is the following result, known as *Mercer's theorem*. This theorem states that if the Gram matrix is positive definite, its eigenvector decomposition can be written as

$$\boldsymbol{K} = \boldsymbol{U}^T \boldsymbol{\Lambda} \boldsymbol{U} \tag{2-32}$$

As derived by Murphy and Liu et al. (Murphy, 2012; Liu et al., 2010), it then follows that each entry of $\boldsymbol{K}$ can be computed as

$$\kappa(\boldsymbol{x}, \boldsymbol{x}') = \phi(\boldsymbol{x})^T \phi(\boldsymbol{x}') =< \phi(\boldsymbol{x}), \phi(\boldsymbol{x}') > \tag{2-33}$$

meaning that the entries of the kernel matrix can be defined by the inner product of some feature vectors that are implicitly defined by the eigenvectors $\boldsymbol{U}$. If the kernel is Mercer, then there exists a function ϕ which maps $\boldsymbol{x} \in \mathcal{X}$ to $\mathbb{R}^D$ such that Equation 2-33 holds. Additionally, ϕ depends on the eigenfunctions of κ, meaning that D is a potentially infinite dimensional space. Additionally, instead of representing feature vectors in terms of kernels $\phi(\boldsymbol{x}) = [\kappa(\boldsymbol{x}, \boldsymbol{x}_1), \dots, \kappa(\boldsymbol{x}, \boldsymbol{x}_N)]$, algorithms can instead work with the input feature vectors $\boldsymbol{x}$ by replacing all inner products $< \boldsymbol{x}, \boldsymbol{x}' >$ with a call to the kernel function $\kappa(\boldsymbol{x}, \boldsymbol{x}')$. This is called the *kernel trick*, and it turns out that many algorithms can be kernelized in this way.

Kernel functions play an important role in the dimensionality reduction literature for both applying nonlinearity to inherently linear problems and in defining similarity measures for

graph-based manifold learning methods. The kernelization of three traditionally linear manifold learning methods, namely: PCA, MDS and LDA, is briefly described in the following.

Kernel PCA. Section 2.2.4.1 showed how PCA could be used to compute linear low-dimensional embeddings of data. This process involved finding the eigenvectors of the empirical data covariance matrix $\hat{\boldsymbol{S}} = \frac{1}{N}\sum_{n=1}^{N} \hat{\boldsymbol{x}}_n \hat{\boldsymbol{x}}_n^T = \frac{1}{N}\hat{\boldsymbol{X}}^T\hat{\boldsymbol{X}}$, where $\hat{\boldsymbol{x}}_n = \boldsymbol{x}_n - \hat{\boldsymbol{\mu}}$ is the mean subtracted feature vector. However, PCA can also be computed by finding the eigenvectors of the inner product matrix $\boldsymbol{X}\boldsymbol{X}^T$ (Murphy, 2012; Wang, 2012). This allows the production of nonlinear embeddings by taking advantage of the kernel trick. This approach is known as Kernel PCA (Schölkopf et al., 1999).

Kernel MDS. (Webb, 2002)

Kernel FDA (KDA). (Ghojogh et al., 2019)

2.2.5.2 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:

$$\boldsymbol{W}_{\boldsymbol{u},\boldsymbol{u}'} = w_{\boldsymbol{u},\boldsymbol{u}'} = \exp\left(-\frac{||\boldsymbol{u} - \boldsymbol{u}'||^2}{\beta}\right) \tag{2-34}$$

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.

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

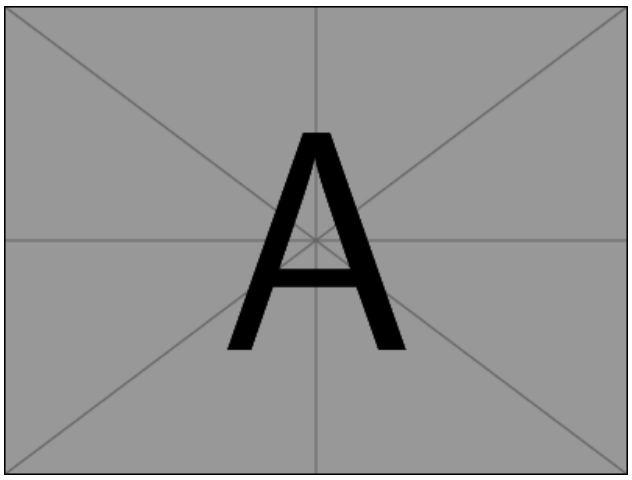


Figure 2-8: Examples of K-nearest neighbor and ϵ-ball graphs

of a K-nearest neighbor graph is depicted in a of Figure 2-8. 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 2-8. 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 $\mathbb{R}^D$. Many dimensionality reduction methods in the literature discover projections of data into a low-dimensional space which preserve topological ordering of the data (Kegl et al., 2008). 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 2-9, 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 2-9, 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 (Sorzano et al., 2014). 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).

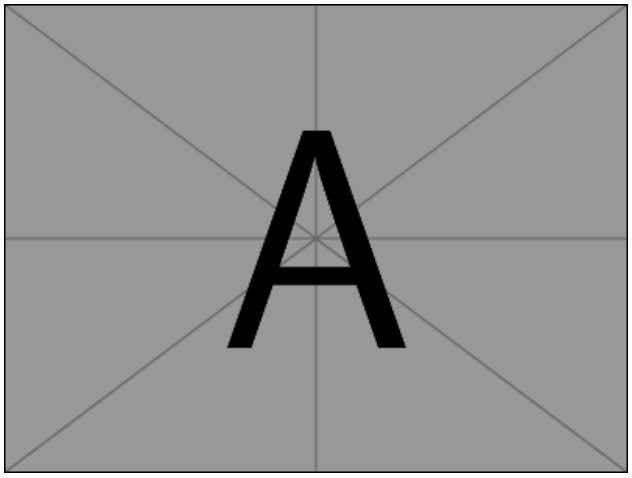


Figure 2-9: Demonstration of geodesic distance

2.2.5.3 General Graph Embedding Framework

2.2.5.4 Isomap

Traditional 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:

$$\boldsymbol{W}_{mn} = w_{mn} = ||\boldsymbol{x}_m - \boldsymbol{x}_n||^2 \tag{2-35}$$

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 (Tenenbaum et al., 2000).

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 financial analysis (Ribeiro et al., 2008), facial and object recognition (Zhang et al., 2018), visualization and classification tasks (Vlachos et al., 2002), 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 (Silva and Tenenbaum, 2003) and looking at conformal maps, as is done in Conformal Isomap (Silva and Tenenbaum, 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.

Supervised Isomap Approaches. 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. The work by Vllachos et al. (Vlachos et al., 2002) was the first to investigate a supervised adaptation of Isomap. Two supervised Isomap procedures were proposed which combine Isomap with a nearest neighbor classifier. These methods, Iso+Ada and WeightedIso take label information into consideration to scale the computed Euclidean distances utilized by Isomap by a constant factor according to class label. The idea is to make points closer in the embedding space if they have the same class label and farther if they have opposing class labels. Ribeiro et al. (Ribeiro et al., 2008) proposed an enhanced supervised Isomap (ES-Isomap) in which the dissimilarity matrix is weighted according to rules which consider class label information. The dissimilarity matrix (considered as the adjacency matrix), $\boldsymbol{W}$, which was the same used in the Supervised Isomap method (X. Geng et al., 2005), is defined as:

$$\boldsymbol{W}(\boldsymbol{x}_m, \boldsymbol{x}_n) = \begin{cases} \sqrt{1 - \exp \frac{-\mathcal{D}^2(\boldsymbol{x}_m, \boldsymbol{x}_n)}{\beta}}, & l_m = l_n \\ \sqrt{\exp \frac{\mathcal{D}^2(\boldsymbol{x}_m, \boldsymbol{x}_n)}{\beta}} - \alpha, & l_m \neq l_n \end{cases} \tag{2-36}$$

where $\mathcal{D}(\boldsymbol{x}_m, \boldsymbol{x}_n)$ denotes the distance measure between samples $\boldsymbol{x}_m$ and $\boldsymbol{x}_n$, β is used to prevent $\boldsymbol{W}(\boldsymbol{x}_m, \boldsymbol{x}_n)$ from increasing too quickly when $\mathcal{D}(\boldsymbol{x}_m, \boldsymbol{x}_n)$ is large and is typically set according to the density of the data, α is a constant in $[0, 1]$ which controls the dissimilarity between points in different classes and keeps the graph from becoming disconnected, and l_m and l_n are the corresponding class labels of samples $\boldsymbol{x}_m$ and $\boldsymbol{x}_n$, respectively. In Equation 2-36, the dissimilarity between two points is greater than or equal to one if their class labels are different and less than 1 if the points have the same class label. Therefore, the between-class

dissimilarity will always be larger than the within-class, which is an important property for classification tasks.

Li and Guo proposed Supervised Isomap with Explicit mapping (SE-Isomap) in (Chun-Guang Li and Jun Guo, 2006). SE-Isomap enforces discriminability on the matrix of geodesic distances, as compared to the Euclidean distance matrix used in the aforementioned approaches, to learn an explicit mapping to the low-dimensional embedding space. Finally, Zhang et al. (Zhang et al., 2018) developed a semi-supervised Isomap to utilize both labeled and unlabeled data points in training. This method aims at minimizing pairwise distances of within-class samples in the same manifold while maximizing the distances over different manifolds.

2.2.5.5 Locally Linear Embedding (LLE)

2.2.5.6 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:

$$\boldsymbol{W}_{mn} = w_{mn} = \exp\left(-\frac{||\boldsymbol{x}_m - \boldsymbol{x}_n||^2}{\beta}\right) \tag{2-37}$$

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

$$J(\boldsymbol{W}, \boldsymbol{Z}) = \frac{1}{2} \sum_{m,n} ||\boldsymbol{z}_m - \boldsymbol{z}_n||^2 w_{mn} = tr(\boldsymbol{Z}^T \boldsymbol{L} \boldsymbol{Z}) \tag{2-38}$$

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:

$$\min_{\boldsymbol{Z}} tr(\boldsymbol{Z}^T \boldsymbol{L} \boldsymbol{Z}) \quad s.t. \quad \boldsymbol{Z}^T \boldsymbol{D} \boldsymbol{Z} = \boldsymbol{I}, \boldsymbol{Z}^T \boldsymbol{L} \boldsymbol{e} = \boldsymbol{0} \tag{2-39}$$

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:

$$\boldsymbol{L}\boldsymbol{v} = \lambda \boldsymbol{D}\boldsymbol{v} \tag{2-40}$$

The column vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_N$ are the solutions of Equation 2-40, 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.

$$N_w(\boldsymbol{x}_n) = \{\boldsymbol{x}_m | l_m = l_n, \exp\left(-\frac{||\boldsymbol{x}_n - \boldsymbol{x}_m||^2}{\beta}\right) > AS(\boldsymbol{x}_n)\} \tag{2-41}$$

$$N_b(\boldsymbol{x}_n) = \{\boldsymbol{x}_m | l_m \neq l_n, \exp\left(-\frac{||\boldsymbol{x}_n - \boldsymbol{x}_m||^2}{\beta}\right) > AS(\boldsymbol{x}_n)\} \tag{2-42}$$

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 2-41 and 2-42 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:

$$W_{w,mn} = \begin{cases} \exp\left(-\frac{||\boldsymbol{x}_n - \boldsymbol{x}_m||^2}{\beta}\right), & \text{if } \boldsymbol{x}_n \in N_w(\boldsymbol{x}_m) \text{ or } \boldsymbol{x}_m \in N_w(\boldsymbol{x}_n) \\ 0, & \text{otherwise} \end{cases} \tag{2-43}$$

$$W_{b,mn} = \begin{cases} 1, \text{ if } \boldsymbol{x}_n \in N_b(\boldsymbol{x}_m) \text{ or } \boldsymbol{x}_m \in N_b(\boldsymbol{x}_n) \\ 0, \text{ otherwise} \end{cases} \tag{2-44}$$

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

$$\boldsymbol{W} = \boldsymbol{W}_w + \boldsymbol{W}_b \tag{2-45}$$

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 $\boldsymbol{Z}$:

$$\min \frac{1}{2} \sum_{m,n} ||\boldsymbol{z}_m - \boldsymbol{z}_n||^2 W_{w,mn} = tr(\boldsymbol{Z}^T \boldsymbol{L}_w \boldsymbol{Z}) \tag{2-46}$$

$$\max \frac{1}{2} \sum_{m,n} ||\boldsymbol{z}_m - \boldsymbol{z}_n||^2 W_{b,mn} = tr(\boldsymbol{Z}^T \boldsymbol{L}_b \boldsymbol{Z}) \tag{2-47}$$

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:

$$\arg\max_{\boldsymbol{Z}} \left\{ \gamma tr(\boldsymbol{Z}^T \boldsymbol{L}_b \boldsymbol{Z}) + (1-\gamma) tr(\boldsymbol{Z}^T \boldsymbol{W}_w \boldsymbol{Z}) \right\} \quad s.t. \quad \boldsymbol{Z}^T \boldsymbol{D}_w \boldsymbol{Z} = \boldsymbol{I} \tag{2-48}$$

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:

$$\arg\max_{\boldsymbol{Z}} \left(\boldsymbol{Z}^T \boldsymbol{B} \boldsymbol{Z} \right) \quad s.t. \quad \boldsymbol{Z}^T \boldsymbol{D}_w \boldsymbol{Z} = \boldsymbol{I} \tag{2-49}$$

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

$$\boldsymbol{B}\boldsymbol{v} = \lambda \boldsymbol{D}_w \boldsymbol{v} \tag{2-50}$$

The column vectors $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_N$ are the generalized eigenvectors of Equation 2-50 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).

2.2.5.7 Hessian Eigenmaps

2.2.5.8 Diffusion Maps

2.2.5.9 Sammon Mapping

2.2.5.10 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

$$p(\boldsymbol{z}_n) = \mathcal{N}(\boldsymbol{z}_n|\boldsymbol{\mu}_0, \boldsymbol{S}_0) \tag{2-51}$$

and the data likelihood is denoted as a multivariate Gaussian

$$p(\boldsymbol{x}_n|\boldsymbol{z}_n, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{W}\boldsymbol{z}_n + \boldsymbol{\mu}, \boldsymbol{\Psi}) \tag{2-52}$$

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:

$$p(\boldsymbol{z}_n) = \mathcal{N}(0, \boldsymbol{I}_d) \tag{2-53}$$

$$p(l_n|\boldsymbol{z}_n) = \text{Ber}(\text{sigm}(\boldsymbol{w}_l^T\boldsymbol{z}_n)) \tag{2-54}$$

$$p(\boldsymbol{x}_n|\boldsymbol{z}_n) = \mathcal{N}(\boldsymbol{W}_x\boldsymbol{z}_n + \boldsymbol{\mu}_x, \sigma^2\boldsymbol{I}_D) \tag{2-55}$$

Discriminative Supervised PCA

2.2.6.2 Generative Topographic Mapping (GTM)

2.2.7 Competitive Hebbian Learning

2.2.8 Deep Learning

Autoencoders.

Graph Convolutional Networks.

M3DNet.

Paper in Papers channel.

2.2.9 UMAP

2.2.10 Stochastic Neighbor Embedding (SNE and t-SNE)

2.2.11 NCA

2.3 Weakly Supervised Manifold Learning and Dimensionality Reduction

Although the specific feature vectors being used in remote sensing applications can be very high-dimensional, the underlying structure of a given dataset set is usually governed by only a few variables. Either implicitly or explicitly, most learning algorithms exploit this underlying structure to make learning and inference possible. If it is available, relevant information for a specific task can generally be incorporated to provide supervision and improve the performance of unsupervised methods. Supervised methods for nonlinear dimensionality reduction assume that the samples lie on a manifold parameterized by multiple latent factors. However, different from traditional manifold learning (where the goal is to preserve the relationships between samples), these methods find the most discriminative low-dimensional representations for classification tasks (Wu, 2015). The optimal embedding uses class label information to minimize distances between nearby points with the same class label while separating samples of different classes. Although supervised manifold learning often outperforms unsupervised methods for classification tasks, this learning cannot be done directly in MIL because of the

uncertainty on the labels (Carbonneau et al., 2016). Moreover, fully-annotated samples are often difficult or impossible to obtain in many remote sensing applications (Zare et al., 2018). Even with the successes of manifold learning, most the of the previous work has mainly focused on either fully supervised or unsupervised learning. Existing work in weakly supervised learning on manifolds has primarily considered the semi-supervised setting (Z. Zhang et al., 2008; Chen et al., 2018; Zhang et al., 2014; Hong et al., 2019; Navaratnam et al., 2007; Stanley III et al., 2018; Tuia and Camps-Valls, 2015; Wang and Mahadevan, 2010, 2011). Methods in this category usually incorporate partially provided image labels and propagate the labels over the manifold approximated by the neighborhood graph on the images. In a broad sense, however, different situations of weak supervision (specifically, Multiple Instance Learning), have not been well studied (Wu, 2015). The existing MIL manifold learning in the literature can be broken into two paradigms: LDA-based approaches and sparse, orthogonal matrix-based techniques (Zhu et al., 2018). Thus, all existing approaches are linear, meaning they may not work well if the underlying bag manifold exhibits curvature. Alternative weakly-supervised dimensionality reduction approaches have been proposed, however, they do not adhere to the constraints of MIL. The current literature for weakly supervised dimensionality reduction is reviewed in the following sections, beginning with reviews of MIL DR approaches, namely: MIDR, MidLABS, CLFDA, MIDA and MI-FEAR. Approaches using alternative definitions of weak learning are addressed at the end of the section.

2.3.1 MIDR

The first true MIL manifold learning experimentation was performed in (Sun et al., 2010) under the orthogonal matrix-based paradigm. 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, traditional 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 et al. proposed *Multiple Instance Dimensionality Reduction* (MIDR), which optimizes an objective through gradient descent to discover sparse, orthogonal projection vectors in the latent space in conjunction with the Multiple Instance Logistic Regression classifier. The goal of MIDR is to discover a projection matrix $\boldsymbol{W} \in \mathbb{R}^{D\times d}$ which will increase discriminability between positive and negative bags in the latent embedding space. If given the k^{th} training bag, $\boldsymbol{B}_k = \{\boldsymbol{x}_{k,1}, \ldots, \boldsymbol{x}_{k,n_k}\}$ with corresponding binary bag-level label $L_k \subset \{0, 1\}$, MIDR attempts to find a matrix $\boldsymbol{W}$ such that the projection of $\boldsymbol{B}_k \subset \mathbb{R}^D$ by $\boldsymbol{W}^T\boldsymbol{B}_k \subset \mathbb{R}^d$ increases the separation between positive and negative bags. The intuition is that the probability of the k^{th} bag being positive $\Pr(L_k = 1|\boldsymbol{W}^T\boldsymbol{B}_k)$ should be close to one if it is positive and close to zero otherwise. This can be achieved by minimizing the squared loss between the actual and predicted label of each bag

$$\min_{\boldsymbol{W}} \sum_{k=1}^{K} (\Pr(L_k = 1|\boldsymbol{W}^T\boldsymbol{B}_k) - L_k)^2 \tag{2-56}$$

Taking advantage of the standard assumption, the posterior probability of a bag can be written in terms of the posterior probabilities of its instances

$$\Pr(L_k = 1|\boldsymbol{W}^T\boldsymbol{B}_k) = \max_n \Pr(l_{k,n} = 1|\boldsymbol{W}^T\boldsymbol{x}_{k,n}) \tag{2-57}$$

Equation 2-56 then becomes

$$\min_{\boldsymbol{W}} \sum_{k=1}^{K} (\max_n \Pr(l_{k,n} = 1|\boldsymbol{W}^T\boldsymbol{x}_{k,n}) - L_k)^2 \tag{2-58}$$

From the objective, it is clear that in order to minimize the squared loss, the distances between the key positive instances and all negative instances should be as large as possible. Additionally, $\boldsymbol{W}$ is required to be orthogonal in order to guarantee the resulting latent features are uncorrelated (to remove redundancy) as well as sparse (to improve interpretability). This implies that the new feature representations of instances $\boldsymbol{x}_{kn}$ in bag $\boldsymbol{B}_k$ are formed by linear combinations of the features in the input feature space. The MIDR optimization problem can

be written succinctly as defined in (Zhu et al., 2018):

$$\min_{\boldsymbol{W},\boldsymbol{\beta}} f(\boldsymbol{W},\alpha) + \gamma||\boldsymbol{W}||_1 \quad \text{s.t.} \quad \boldsymbol{W}^T\boldsymbol{W} = \mathbb{I}_d \tag{2-59}$$

where γ is a positive number which controls the balance between the sparsity term $||\boldsymbol{W}||_1$ and the fitting term $f(\boldsymbol{W},\alpha) = \sum_{k=1}^{K}(P_k(\boldsymbol{W},\boldsymbol{\beta}) - L_k)^2$. In this case,

$$\begin{aligned} P_k(\boldsymbol{W},\boldsymbol{\beta}) &= softmax_\alpha(P_{k,1}(\boldsymbol{W},\boldsymbol{\beta}),\ldots,P_{k,n_k}(\boldsymbol{W},\boldsymbol{\beta})) \\ &= \frac{\sum_{n=1}^{n_k} P_{k,n} e^{\alpha P_{k,n}(\boldsymbol{W},\boldsymbol{\beta})}}{\sum_{n=1}^{n_k} e^{\alpha P_{k,n}(\boldsymbol{W},\boldsymbol{\beta})}} \end{aligned} \tag{2-60}$$

is the softmax approximation over n of $\max(P_{k,1}(\boldsymbol{W},\boldsymbol{\beta}),\ldots,P_{k,n_k}(\boldsymbol{W},\boldsymbol{\beta}))$. A popular way to estimate the posterior probability is by logistic regression

$$P_{k,n}(\boldsymbol{W},\boldsymbol{\beta}) = \Pr(l_{k,n} = 1|\boldsymbol{W}^T,\boldsymbol{x}_{k,n}) = \frac{1}{1+\exp(-\boldsymbol{\beta}^T\boldsymbol{W}^T\boldsymbol{x}_{k,n})} \tag{2-61}$$

Pseudo-code for MIDR is provided in Algorithm 2.

Algorithm 2 MIDR

Input: Multiple-instance dataset $\boldsymbol{B} = \{\boldsymbol{B}_1,\ldots,\boldsymbol{B}_K\}$, $\boldsymbol{L} = \{L_1,\ldots,L_K\}$, $L_k \in \{-1,+1\}$, sparsity parameter γ
Output: Projection matrix $\boldsymbol{W}$

```
while Not converged do
    Train multiple instance logistic regression h using data W^T B, L
    for B_k ∈ B do
        P_k ← probability h(B_k)
    end for
    Optimize Equation 2-59 for new W
end while
```

In (Sun et al., 2010), gradient descent was used to optimize the objective. An alternating optimization scheme was employed that switched between estimating the parameters of the MI Logistic Regression and solving for a new sparse, orthogonal embedding matrix. It was found that the newly developed method outperformed unsupervised instance-level dimensionality reduction approaches (applied to bags) for bag-level classification. MIDR was later revisited by Zhu et al. where the optimization problem was reformulated using the *intertial proximal*

alternating linearized minimization (iPALM) method (Zhu et al., 2018). The advantage of *Multiple Instance Augmented Lagrangian Multiplier* (MI-ALM) this approach is that the problem variables can be managed sperately and updated effectively. Additionally, the global convergence of MIDR was proved.

2.3.2 MidLABS

The other existing approaches for dimensionality reduction with multiple instance learning follow a LDA scheme. *Multi-Instance Dimensionality reduction by Learning a mAximum Bag margin Subspace* (MidLABS) (Ping et al., 2010) applies LDA to find a projection vector which simultaneously maximizes between-class scattering and minimizes within-class scattering to separate positive and negative bags in the embedding space. While most MIL approaches assume instances are independently and identically distributed (IID), MidLABS represents each bag as a neighborhood graph in order to take advantage of data structure and jointly constructs the scatter matrices by evaluating the scattering between bags. MidLABS optimizes the following objective:

$$\max_{\boldsymbol{w}} J(\boldsymbol{w}) = \max_{\boldsymbol{w}} \frac{\boldsymbol{w}^T(\sum_{L_i \neq L_j} \boldsymbol{K}_{ij})\boldsymbol{w}}{\boldsymbol{w}^T(\sum_{L_i = L_j} \boldsymbol{K}_{ij})\boldsymbol{w}} \tag{2-62}$$

By defining a bag distance measure, $\boldsymbol{K}$, the the between-class and within-class scatter matrices can be constructed as

$$\boldsymbol{S}_b = \sum_{L_i \neq L_j} \boldsymbol{K}_{ij} \tag{2-63}$$

and

$$\boldsymbol{S}_w = \sum_{L_i = L_j} \boldsymbol{K}_{ij} \tag{2-64}$$

It can be observed that this problem follows LDA exactly, and can thus be solved as the generalized eigenvalue problem:

$$\max_{\boldsymbol{w}} J(\boldsymbol{w}) = \max_{\boldsymbol{w}} \frac{\boldsymbol{w}^T \boldsymbol{S}_b \boldsymbol{w}}{\boldsymbol{w}^T \boldsymbol{S}_w \boldsymbol{w}} \tag{2-65}$$

In order to take structural information into account, the customized bag distance measurement is defined by:

$$\boldsymbol{K}_{ij} = \frac{\sum_{a=1}^{n_i}\sum_{b=1}^{n_j}(\boldsymbol{x}_{ia}-\boldsymbol{x}_{jb})(\boldsymbol{x}_{ia}-\boldsymbol{x}_{jb})^T}{n_i n_j} + C\frac{\sum_{c=1}^{m_i}\sum_{d=1}^{m_j}(\boldsymbol{e}_{ic}-\boldsymbol{e}_{jd})(\boldsymbol{e}_{ic}-\boldsymbol{x}_{jd})^T}{n_i^2 n_j^2} \quad (2\text{-}66)$$

where $\boldsymbol{x}_{ia}$ is the a^{th} instance in the i^{th} bag, n_i is the total number of instances in the i^{th} bag, $\boldsymbol{e}_{ic}$ is the c^{th} edge in the i^{th} bag and m_i is the total number of edges in the i^{th} bag. This bag distance measurement represents each bag by as an ϵ-graph, where each instance is treated as a node. An edge exists between two nodes if the Euclidean distance between them is lower than a threshold, ϵ. (Latham, 2015). This measurement compares the closeness of bags by summing over all pairs of instances between the bags. Pseudo-code for MidLABS is provided in Algorithm 3.

Algorithm 3 MidLABS

Input: Multiple-instance dataset $\{\boldsymbol{B}_1,\ldots,\boldsymbol{B}_K\}$, $\{L_1,\ldots,L_K\}$, $L_k \in \{-1,+1\}$
Output: Linear projection vector $\boldsymbol{w}$

1: **for** bag $\boldsymbol{B}_k$ in $\{\boldsymbol{B}_1,\ldots,\boldsymbol{B}_K\}$ **do**
2: $\quad G_k \leftarrow \epsilon$-graph for $\boldsymbol{B}_k$
3: **end for**
4: Define $\boldsymbol{K}_{ij} = \frac{\sum_{a=1}^{n_i}\sum_{b=1}^{n_j}(\boldsymbol{x}_{ia}-\boldsymbol{x}_{jb})(\boldsymbol{x}_{ia}-\boldsymbol{x}_{jb})^T}{n_i n_j} + C\frac{\sum_{c=1}^{m_i}\sum_{d=1}^{m_j}(\boldsymbol{e}_{ic}-\boldsymbol{e}_{jd})(\boldsymbol{e}_{ic}-\boldsymbol{x}_{jd})^T}{n_i^2 n_j^2}$
5: $\boldsymbol{S}_b = \sum_{L_i \neq L_j} \boldsymbol{K}_{ij}$
6: $\boldsymbol{S}_w = \sum_{L_i = L_j} \boldsymbol{K}_{ij}$
7: Solve the generalized eigenvalue problem $\boldsymbol{S}_b\boldsymbol{w} = \lambda\boldsymbol{S}_w\boldsymbol{w}$
8: Sort eigenvectors $\boldsymbol{w}$ by their eigenvalues λ
9: $\boldsymbol{w} \in \mathbb{R}^{D\times 1} \leftarrow$ top eigenvector

2.3.3 MIDA

In 2014, Chia et al. introduced Multiple-Instance Discriminant Analysis (MIDA) (Chai et al., 2014). MIDA has the same objective as MidLABS, which is to discover a linear projection basis which separates bags in the embedding space. Both MIDA and MidLABS can be considered as MI extensions of LDA. However, the way these algorithms construct their scatter matrices is very different. While MidLABS constructs its scatter matrices at the bag level by directly evaluating the scattering amongst bags, MIDA uses instance-level information to

formulate the within-class and between-class scatter. In other words, MIDA selects a prototype for each bag and utilizes the selected instances as bag representatives for constructing the scatter. The mean of all negative instances is used as the negative class prototype. The difficulty with this approach is the ambiguity on which instances are truly positive. The other major difference between MidLABS and MIDA is that MIDA does not consider structural information of data when formulating the scatter matrices. To select candidate positive prototypes, MIDA initializes a set by selecting the most-likely positive instances as those with the lowest density in a Gaussian likelihood estimated by all instances in the negative bags. An iterative optimization procedure is applied which trades-off between candidate positive instance selection and learning the projection weights into the latent space which maximizes the separation between bags with different labels and minimizes the distances between bags with the same label. Pseudo-code for MIDA is provided in Algorithms 4 and 5.

Algorithm 4 Initialization of MIDA

Input: Multiple-instance dataset $\boldsymbol{B} = \{\boldsymbol{B}_1, \ldots, \boldsymbol{B}_K\}$, $\{L_1, \ldots, L_K\}$, $L_k \in \{-1, +1\}$, parameter β

Output: Initialized positive prototypes $\boldsymbol{x}^+ = \{\boldsymbol{x}_1^+, \ldots, \boldsymbol{x}_{N^+}^+\}$

1: **for** bag $\boldsymbol{B}_k$ in $\boldsymbol{B}^+$ **do**

2: $\quad \boldsymbol{x}_k^+ \leftarrow \arg\min_{\boldsymbol{x}_{kn}} C \sum_{m=1}^{N^-} \sum_{o=1}^{N_m^-} \exp\left(\frac{||\boldsymbol{x}-\boldsymbol{x}_{mo}^-||_2^2}{\beta}\right) \forall n = 1, \ldots, N_k^+$

3: **end for**

Algorithm 5 Projection vector calculation process of MIDA

Input: Multiple-instance dataset $\{\boldsymbol{B}_1, \ldots, \boldsymbol{B}_K\}$, $\{L_1, \ldots, L_K\}$, $L_k \in \{-1, +1\}$, Initialized positive prototypes $\boldsymbol{x}^+ = \{\boldsymbol{x}_1^+, \ldots, \boldsymbol{x}_{N^+}^+\}$

Output: Linear projection vector $\boldsymbol{w}$

1: $\boldsymbol{\mu}^+ \leftarrow \frac{1}{N^+} \sum_{k=1}^{N^+} \boldsymbol{x}_k^+$

2: $\boldsymbol{\mu}^- \leftarrow C \sum_{m=1}^{N^-} \sum_{o=1}^{N_m^-} \boldsymbol{x}_{mo}^-$

3: $\boldsymbol{S}_w^+ \leftarrow \sum_{k=1}^{N^+} (\boldsymbol{x}_k^+ - \boldsymbol{\mu}^+)(\boldsymbol{x}_k^+ - \boldsymbol{\mu}^+)^T$

4: $\boldsymbol{S}_w^- \leftarrow \sum_{m=1}^{N^-} (\boldsymbol{x}_m^- - \boldsymbol{\mu}^-)(\boldsymbol{x}_m^- - \boldsymbol{\mu}^-)^T$

5: $\boldsymbol{S}_w \leftarrow \boldsymbol{S}_w^+ + \boldsymbol{S}_w^-$

6: $\boldsymbol{S}_b \leftarrow \sum_{k=1}^{N^+} \sum_{m=1}^{N^-} (\boldsymbol{x}_k^+ - \boldsymbol{x}_m^-)(\boldsymbol{x}_k^+ - \boldsymbol{x}_m^-)^T$

7: Solve the generalized eigenvalue problem $\boldsymbol{S}_b \boldsymbol{w} = \lambda \boldsymbol{S}_w \boldsymbol{w}$

8: Sort eigenvectors $\boldsymbol{w}$ by their eigenvalues λ

9: $\boldsymbol{w} \in \mathbb{R}^{D \times 1} \leftarrow$ top eigenvector of $\frac{\boldsymbol{w}^T \boldsymbol{S}_b \boldsymbol{w}}{\boldsymbol{w}^T \boldsymbol{S}_w \boldsymbol{w}}$

2.3.4 CLFDA

Finally, Citation Local Fisher Linear Discriminant Analysis (CLFDA) (Saehoon Kim and Seungjin Choi, 2010) incorporates citation and reference information into local Fisher Discriminant Analysis, thus it can also be treated as a MI extension to LDA. Contrary to the previously-mentioned approaches, CLFDA operates at the instance-level, attempting to find a subspace where positive and negative instances are maximally-separable (Latham, 2015). CLFDA can be viewed as complimentary to MIDA (Chai et al., 2014). Whereas, MIDA tries to seek true positive instances in positive bags, CLFDA attempts to detect incorrectly labeled instances (false positive) in positive bags. CLFDA pre-labels all instances with the label of their bag $l_{kn} = L_k, \forall n = 1, \ldots, n_k$, then utilizes neighborhood information to detect false positive instances. In other words, it uses the assumption that if an instance in a positive bag is close to many points in the negative bags, it is likely also negative. CLFDA defines *references* simply as the nearest neighbors to $\boldsymbol{x}_n$, while $citers$ are defined as the samples which have $\boldsymbol{x}_n$ in their C-nearest neighbors, or $citers(\boldsymbol{x}_n) = \{\boldsymbol{x}_m | \boldsymbol{x}_n \in CNN(\boldsymbol{x}_m), m = 1, \ldots, N\}$. The steps of CLFDA are to 1.) construct a $\max(R, C)$-NN graph, where the $\max(R, C)$-NN graph is a K-NN graph with $K = \max(R, C)$. This is used to detect false positives and re-label them as negative. If the ratio of instances from negative bags versus instances from positive bags in the references and citers of $\boldsymbol{x}_n$ exceeds a threshold τ, then $\boldsymbol{x}_n$ is given a negative label. This is only done for instances from positive bags, as the SMI assumptions states that all instances in negative bags should be negative. 2.) Construct scatter matrices using the provided positive and negative instances. 3.) Find projection weights which simultaneously maximize the distances between positive and negative bags and minimizes the distances between bags with the same class labels using *Local Fisher Discriminant Analysis* (LFDA). LFDA is a version of LDA designed to address multi-modal data distributions. The primary difference between LDA and LFDA is that when maximizing and minimizing the between-class and within-class scatter matrices, respectively, the scatter contribution of a single instance pair is weighted by the locality of the pair. This prevents instances from different clusters sharing the same label

from being forced into the same space. This also allows multiple dimensions to be analyzed in the embedding space, whereas LDA is limited to a single dimension for binary classification. Pseudo-code for CLFDA is given in Algorithm 6 (Latham, 2015). A potential downside of CLFDA, however, is that it has been shown that pre-labeling all instances in positive bags as positive often leads to poor results, especially when there are large numbers of negative instances in the positive bags (Chai et al., 2014).

Algorithm 6 CLFDA

Input: Multiple-instance dataset $\boldsymbol{B} = \{\boldsymbol{B}_1, \ldots, \boldsymbol{B}_K\}$, $\{L_1, \ldots, L_K\}$, $L_k \in \{-1, +1\}$, parameters C, R, τ

Output: Projection matrix $\boldsymbol{W}$

$\boldsymbol{X} \leftarrow$ instances in $\boldsymbol{B}$
$G \leftarrow \max(R, C)$-nearest neighbor graph of $\boldsymbol{X}$
for $n = 1 \rightarrow N$ **do**
    $R_n \leftarrow R$-nearest references of instance $\boldsymbol{x}_n$
    $C_n \leftarrow C$-nearest citers of instance $\boldsymbol{x}_n$
    $N_n^- \leftarrow$-number of instances from negative bags in $R_n + C_n$
    $N_n^+ \leftarrow$-number of instances from positive bags in $R_n + C_n$
    if $\frac{N_n^-}{N_n^+} \geq \tau$ or $\boldsymbol{x}_n$ is from a negative bag **then**
        instance label $l_n \leftarrow -1$
    else
        $l_n \leftarrow +1$
    end if
end for
$\boldsymbol{A}^b \leftarrow$ between-class affinity matrix
$\boldsymbol{A}^w \leftarrow$ within-class affinity matrix
Between-class scatter matrix $\boldsymbol{S}_b = \frac{1}{2} \sum_{m,n=1}^{N} \boldsymbol{A}_{mn}^b (\boldsymbol{x}_m - \boldsymbol{x}_n)(\boldsymbol{x}_m - \boldsymbol{x}_n)^T$
Within-class scatter matrix $\boldsymbol{S}_w = \frac{1}{2} \sum_{m,n=1}^{N} \boldsymbol{A}_{mn}^w (\boldsymbol{x}_m - \boldsymbol{x}_n)(\boldsymbol{x}_m - \boldsymbol{x}_n)^T$
$\boldsymbol{W} \in \mathbb{R}^{D \times d} \leftarrow$ top d eigenvectors of $\frac{\boldsymbol{w}^T \boldsymbol{S}_b \boldsymbol{w}}{\boldsymbol{w}^T \boldsymbol{S}_w \boldsymbol{w}}$

As previously mentioned, all existing MIL dimensionality reduction approaches in the literature are solely linear. However, many modalities exhibit nonlinear variation. Therefore, nonlinear manifold learning approaches should be developed which operate under the multiple instance learning framework.

2.3.5 MI-FEAR

While the previously mentioned approaches were based on feature extraction, multiple instance dimensionality reduction has also been investigated under the feature selection paradigm. Specifically, Latham used feature ranking to determine the most important features for instance-level classification (Latham, 2015). Feature ranking considers each feature independently according to a scoring function, thus revealing properties of the individual features by which they may be compared and ranked. The challenge of learning a good feature ranking under an instance-space metric is that the instance labels are not available under the MIL framework. *Multiple-Instance Feature Ranking* (MI-FEAR) assumes one-sided noise by giving every instance the label of its corresponding bag. This essentially transforms the weakly-supervised problem into a supervised one. Pseudo-code for MI-FEAR is provided in Algorithm 7.

Algorithm 7 MI-FEAR

Input: Multiple-instance dataset $\boldsymbol{B} = \{\boldsymbol{B}_1, \ldots, \boldsymbol{B}_K\}$, $\boldsymbol{L} = \{L_1, \ldots, L_K\}$, $L_k \in \{-1, +1\}$, evaluation metric $\mathcal{L}$, learning algorithm $\mathcal{A}$, dimensionality of reduced-dimensional space d
Output: $[\theta_i$ for θ_i in $V[1, \ldots, d]]$

```
X, Y^γ ← supervised dataset of bag-labeled instances using (B, L)
V ← [ ]
for feature θ_i in feature set Θ do
    X^θ ← X with all features, excluding θ_i
    h_θ ← output of A when trained on input (X^θ, Y^γ)
    V[i] ← performance of h_θ on input (X^θ, Y^γ) according to L, θ_i
end for
V ← SortDecreasing(V)
```

Each feature θ, is given a score based on the performance of a learned hypothesis h_θ operating on a single feature. The performance is determined an evaluation metric $\mathcal{L}$. Once a score has been assigned to every feature, the features are ranked according to relative importance and the top d features are retained as the feature set. As stated in Section 2.3.4, the characteristic accuracy of MI-FEAR is based on noisy labels, where each instance is given its corresponding bag-level label. This approach has proven non-ideal in the literature for determining accurate instance-level classification. However, a benefit of feature selection is that, unlike feature

extraction, the reduced-dimensional space retains its interpret-ability. In other words, the representations of individual features do not change, meaning they keep their real-world relevance, if applicable. On the other hand, feature selection requires that a subset of useful features for classification already exist in the training set. Empirical results showed that consistently achieved lower instance-level classification accuracy than CLFDA and MidLABS, but had a significantly lower run-time.

2.3.6 Comparison Table of MI Dimensionality Reduction Methods

Table 2-1 shows a comparison between the multiple instance dimensionality reduction methods reviewed in Section 2.3.

Table 2-1: Summary of multiple instance dimensionality reduction approches.

Multiple Instance Dimensionality Reduction			
Method	Summary	Reduction Method	Classification Level
MIDR	Finds a sparse, orthogonal linear projection matrix optimized for bag-level logistic regression	Linear, orthogonal projection	bag-level
MidLABS	Finds linear projection vector using LDA defined from bag-similarity kernel	LDA	bag-level
MIDA	Learns linear projection vector using LDA defined from bag representative vectors	LDA	instance-level
CLFDA	Learns linear projection matrix using local discriminant analysis defined from instance scatter. Instance labels are provided as bag labels and refined.	LFDA	instance-level
MI-FEAR	Incrementally leaves out feature and evaluates performance loss to provide feature score	Feature selection	instance-level

2.3.7 General Weak Supervision

Alternative approaches to weakly supervised dimensionality reduction that do not follow the MIL framework have also been proposed. For example, Wu studied weakly supervised manifold learning for manifold factorization using image-level labels, where the labels were the variation of interest. The primary idea was to use weak labels to find image pairs that should

be more similar after removing unwanted image variation. Wu used an alignment method constrained by Hessian regularization to learn a manifold regression, such that new test images would be projected smoothly into a space near neighboring input images (Wu, 2015). Gaur et al. used weakly supervised manifold learning to perform dense semantic object correspondence (Gaur and Manjunath, 2017). The objective of the semantic object correspondence problem is to compute dense association maps for a pair of images such that the same object parts get matched, even for very differently appearing object instances. The goal of Gaur's work was to learn a manifold sucht that features belonging to the same semantic object parts were projected closer to each other on the manifold. This was achieved by re-purposing deep convolutional features from a classification network, where the labels were weak segmentations of object parts. These features were then projected onto a manifold using LDA. *Hierarchical Agglomerative Clustering* (HAC) was performed in the embedding space and the labels were refined with respect to geodesic distance on the manifold. This method was inspired by the *manifold assumption*, which states that similar objects (even though disparate in the input feature space), should share an intrinsic manifold. In this way, feature embeddings are learned by an optimization process which is rewarded for projecting features closer on the manifold if they have low feature-space dissimilarity. Additionally, the optimization penalizes feature clusters whose geometric structure is inconsistent with the observed geometric structures of object parts.

An alternative approach for discriminative dimensionality reduction with weak labels is through the application of metric embedding, which is discussed in detail in Section 2.4. While they can be fully supervised, metric embedding techniques often consider groups of samples (usually two or three), jointly, to learn an embedding function. Under this type of weak supervision, only a notion of semantic similarity is needed, as compared to direct class labels.

2.4 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 2.4.1. 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: $\mathcal{D}(\boldsymbol{x}, \boldsymbol{z}) \leq \mathcal{D}(\boldsymbol{x}, \boldsymbol{y}) + \mathcal{D}(\boldsymbol{y}, \boldsymbol{z})$

The non-negativity rule states that the metric evaluated on two instances must have a positive value or be equal to zero. From the Identity rule, we can see that the metric may only be defined as zero if the two instances being evaluated are exactly the same (thus the dissimilarity is zero). The Symmetry rule states that a metric evaluated between two instances must be the same regardless of ordering. Finally, the Triangle Inequality says, intuitively, that the direct distance between two instances $\boldsymbol{x}$ and $\boldsymbol{z}$ is smaller than the distance between $\boldsymbol{x}$ and $\boldsymbol{y}$ plus the distance between $\boldsymbol{y}$ and $\boldsymbol{z}$. Both sides of the inequality will be equal if and only if $\boldsymbol{y}$ lies on the path between $\boldsymbol{x}$ and $\boldsymbol{z}$ on which the metric is defined.

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 $\mathcal{D}(\boldsymbol{x}, \boldsymbol{y}) : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ 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.

2.4.0.1 Metric Learning

Thorstensen thesis, Raducanu S-LE

2.4.0.2 Preference Learning

Choquet integral Can we use to preference learning to discern instances which are likely to be positive and negative, then use these rankings for embedding?

2.4.1 Ranking Loss

2.4.1.1 Pairwise Loss

2.4.1.2 Contrastive Loss

Definition of Contrastive Loss:

2.4.1.3 Triplet Loss

Definition of Triplet Loss:

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

2.4.1.4 Large-Margin K-Nearest Neighbors (LMNN)

2.4.1.5 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).

$$||f(x_i^a) - f(x_i^p)||_2^2 + \alpha < ||f(x_i^a) - f(x_i^n)||_2^2, \quad \forall (f(x_i^a), f(x_i^p), f(x_i^n)) \in \mathcal{T} \tag{2-67}$$

$$\mathcal{L} = ||f(x_i^a) - f(x_i^p)||_2^2 - ||f(x_i^a) - f(x_i^n)||_2^2 + \alpha \tag{2-68}$$

2.4.1.6 Siamese Neural Networks

2.4.2 Weakly Supervised Dimensionality Reduction with Metric Embedding

Xu et al. proposed a weakly supervised dimensionality reduction approach based on the large-margin principle in which the angle between preference pairs is consistent while the distance between examples in preference pairs is maximized (Xu et al., 2014).

CHAPTER 3
PROBLEM DESCRIPTION

CHAPTER 4
EXPERIMENTAL DESIGN

Wei and Zhou (2016)

CHAPTER 5
PRELIMINARY WORK

The work by Wei et al. in Wei and Zhou (2016) suggested that for image classification tasks, certain formulations of MIL were better suited than others. Algorithms such as miGraph, MIBoosting and miFV which assume non-i.i.d samples or take advantage of aggregating properties of bags tend to work better than those which adopt the standard assumption. The authors of this work recommend miGraph with LBP bag generation or MIBoosting with Single Blob generation for image classification. Additionally, classification performance tended to increase as the number of instances increased.

CHAPTER 6
FUTURE TASKS

CHAPTER 7
CONCLUSIONS

REFERENCES

Fabio Aiolli and Alessandro Sperduti. Learning preferences for multiclass problems. 01 2004.

Jaume Amores. Multiple instance classification: Review, taxonomy and comparative study. *Artificial Intelligence*, 201:81 – 105, 2013. ISSN 0004-3702. doi: https://doi.org/10.1016/j.artint.2013.06.003.

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.

Amina Asif, Wajid Arshad Abbasi, Farzeen Munir, Asa Ben-Hur, and Fayyaz ul Amir Afsar Minhas. pylemmings: Large margin multiple instance classification and ranking for bioinformatics applications, 2017.

Boris Babenko, Zhuowen Tu, and Serge J. Belongie. Simultaneous learning and alignment: Multi-instance and multi-pose learning. 2008.

Boris Babenko, Nakul Verma, Piotr Dollr, and Serge Belongie. Multiple instance learning with manifold bags. pages 81–88, 01 2011.

G. Baudat and F. Anouar. Generalized discriminant analysis using a kernel approach. *Neural Computation*, 12(10):2385–2404, 2000. doi: 10.1162/089976600300014980.

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. Bergeron, G. Moore, J. Zaretzki, C. M. Breneman, and K. P. Bennett. Fast bundle algorithm for multiple-instance learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(6):1068–1079, June 2012. ISSN 1939-3539. doi: 10.1109/TPAMI.2011.194.

Charles Bergeron, Jed Zaretzki, Curt Breneman, and Kristin P. Bennett. Multiple instance ranking. In *Proceedings of the 25th International Conference on Machine Learning*, ICML 08, page 4855, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605582054. doi: 10.1145/1390156.1390163. URL https://doi.org/10.1145/1390156.1390163.

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.

J. Bocinsky, C. H. McCurley, D. Shats, and A. Zare. Investigation of initialization strategies for the multiple instance adaptive cosine estimator. In *Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XXIV, 110120N*, volume 11012 of *Proc.SPIE*, May 2019. doi: 10.1117/12.2519463.

L. Cao, F. Luo, L. Chen, S. Yihan, H. Wang, C. Wang, and R. Ji. Weakly supervised vehicle detection in satellite images via multi-instance discriminative learning. *Pattern Recognition*, 64, 12 2016. doi: 10.1016/j.patcog.2016.10.033.

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. Chen, J. Wang, X. Li, and X. Sun. Robust semi-supervised manifold learning algorithm for classification. *Mathematical Problems in Engineering*, 2018:1–8, 02 2018. doi: 10.1155/2018/2382803.

V. Cheplygina, M. Bruijne, and J. P. W. Pluim. Not-so-supervised: A survey of semi-supervised, multi-instance, and transfer learning in medical image analysis. *Medical Image Analysis*, 54:280 – 296, 2019. ISSN 1361-8415. doi: https://doi.org/10.1016/j.media.2019.03.009.

Wei Chu and Zoubin Ghahramani. Preference learning with gaussian processes. In *Proceedings of the 22nd International Conference on Machine Learning*, ICML 05, page 137144, New York, NY, USA, 2005. Association for Computing Machinery. ISBN 1595931805. doi: 10.1145/1102351.1102369.

Chun-Guang Li and Jun Guo. Supervised isomap with explicit mapping. In *First International Conference on Innovative Computing, Information and Control - Volume I (ICICIC'06)*, volume 3, pages 345–348, Aug 2006. doi: 10.1109/ICICIC.2006.530.

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

M. Cook, A. Zare, and D. K. C. Ho. Buried object detection using handheld wemi with task-driven extended functions of multiple instances. In *Proc. SPIE 9823, Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XXI, 98230A*, volume 9823 of *Proc. SPIE*, pages 9823 – 9823 – 9, Apr. 2016. doi: 10.1117/12.2223349.

Ofer Dekel, Yoram Singer, and Christopher D Manning. Log-linear models for label ranking. In S. Thrun, L. K. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*, pages 497–504. MIT Press, 2004.

Thomas Deselaers and Vittorio Ferrari. A conditional random field for multiple-instance learning. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML10, page 287294, Madison, WI, USA, 2010. Omnipress. ISBN 9781605589077.

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.

K. Fukunaga and J. M. Mantock. Nonparametric discriminant analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-5(6):671–678, Nov 1983. ISSN 1939-3539. doi: 10.1109/TPAMI.1983.4767461.

Johannes Fürnkranz and Eyke Hüllermeier. Pairwise preference learning and ranking. In Nada Lavrač, Dragan Gamberger, Hendrik Blockeel, and Ljupčo Todorovski, editors, *Machine Learning: ECML 2003*, pages 145–156, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.

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.

M. Gao, A. Li, R. Yu, V. I. Morariu, and L. S. Davis. C-WSL: count-guided weakly supervised localization. *CoRR*, abs/1711.05282, 2017. URL http://arxiv.org/abs/1711.05282.

Thomas Gärtner, Peter A. Flach, Adam Kowalczyk, and Alexander J. Smola. Multi-instance kernels. In *ICML*, 2002.

U. Gaur and B. S. Manjunath. Weakly supervised manifold learning for dense semantic object correspondence. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1744–1752, Oct 2017. doi: 10.1109/ICCV.2017.192.

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

Shabnam Ghaffarzadegan. Deep multiple instance feature learning via variational autoencoder. In *AAAI Workshops*, 2018.

Benyamin Ghojogh, Fakhri Karray, and Mark Crowley. Fisher and kernel fisher discriminant analysis: Tutorial. *CoRR*, abs/1906.09436, 2019. URL https://arxiv.org/abs/1906.09436.

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

H. Hajimirsadeghi and G. Mori. Multi-instance classification by max-margin training of cardinality-based markov networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(9):1839–1852, Sep. 2017. ISSN 0162-8828. doi: 10.1109/TPAMI.2016.2613865.

D. Hong, N. Yokoya, N. Ge, J. Chanussot, and X. X. Zhu. Learnable manifold alignment (lema): A semi-supervised cross-modality learning framework for land cover and land use classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 147:193 – 205, 2019. ISSN 0924-2716.

Yang Hu, Mingjing Li, and Nenghai Yu. Multiple-instance ranking: Learning to rank images for image retrieval. pages 1–8, 07 2008. ISBN 978-1-4244-2242-5. doi: 10.1109/CVPR.2008.4587352.

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.

M. Ilse, Jakub M. Tomczak, and M. Welling. Attention-based deep multiple instance learning. *CoRR*, abs/1802.04712, 2018. URL http://arxiv.org/abs/1802.04712.

C. Jiao. *Target Concept Learning From Ambiguously Labeled Data*. PhD thesis, Univ. of Missouri, Columbia, MO, Dec. 2017.

Changzhe Jiao, Chao Chen, Ronald G. McGarvey, Stephanie Bohlman, Licheng Jiao, and Alina Zare. Multiple instance hybrid estimator for hyperspectral target characterization and sub-pixel target detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 146:235 – 250, 2018. ISSN 0924-2716. doi: https://doi.org/10.1016/j.isprsjprs.2018.08.012.

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.

A. C. Latham. Multiple-instance feature ranking. Master's thesis, Case Western Reserve University, Cleveland, OH, August 2015.

Christian Leistner, Amir Saffari, and Horst Bischof. Miforests: Multiple-instance learning with randomized trees. In Kostas Daniilidis, Petros Maragos, and Nikos Paragios, editors, *Computer Vision – ECCV 2010*, pages 29–42, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

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.

Shaohua Li, Yong Liu, Xiuchao Sui, Cheng Chen, Gabriel Tjio, Daniel Shu Wei Ting, and Rick Siow Mong Goh. Multi-instance multi-scale cnn for medical image classification. *Medical Image Computing and Computer Assisted Intervention MICCAI 2019*, page 531539, 2019.

ISSN 1611-3349. doi: 10.1007/978-3-030-32251-9_58. URL http://dx.doi.org/10.1007/978-3-030-32251-9_58.

Weifeng Liu, Jose C. Principe, and Simon Haykin. *Kernel Adaptive Filtering: A Comprehensive Introduction*. Wiley Publishing, 1st edition, 2010. ISBN 0470447532.

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.

O. Maron and T. Lozano-Pérez. A framework for multiple-instance learning. In *Proceedings of the 1997 Conference on Advances in Neural Information Processing Systems 10*, NIPS '97, pages 570–576, Cambridge, MA, USA, 1998. MIT Press. ISBN 0-262-10076-2.

O. Maron and A. L. Ratan. Multiple-instance learning for natural scene classification. In *Proceedings of the Fifteenth International Conference on Machine Learning*, ICML '98, pages 341–349, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc. ISBN 1-55860-556-8.

C. H. McCurley, J. Bocinsky, and A. Zare. Comparison of hand-held wemi target detection algorithms. In *Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XXIV, 110120U*, volume 11012 of *Proc.SPIE*, May 2019. doi: 10.1117/12.2519454.

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.

R. Navaratnam, A. W. Fitzgibbon, and R. Cipolla. The joint manifold model for semi-supervised multi-valued regression. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8, Oct 2007. doi: 10.1109/ICCV.2007.4408976.

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.

Wei Ping, Ye Xu, Ren Kexin, Chi-Hung Chi, and Shen Furao. Non-i.i.d. multi-instance dimensionality reduction by learning a maximum bag margin subspace. volume 1, 01 2010.

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.

Bernardete Ribeiro, Armando Vieira, and João Carvalho das Neves. Supervised isomap with dissimilarity measures in embedding learning. In José Ruiz-Shulcloper and Walter G. Kropatsch, editors, *Progress in Pattern Recognition, Image Analysis and Applications*, pages 389–396, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

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.

José Francisco Ruiz-Muñoz, Mauricio Orozco-Alzate, and Germán Castellanos-Domínguez. Multiple instance learning-based birdsong classification using unsupervised recording

segmentation. In *IJCAI*, 2015.

Saehoon Kim and Seungjin Choi. Local dimensionality reduction for multiple instance learning. In *2010 IEEE International Workshop on Machine Learning for Signal Processing*, pages 13–18, Aug 2010. doi: 10.1109/MLSP.2010.5589175.

Bernhard Schölkopf, Alexander J. Smola, and Klaus-Robert Müller. *Kernel Principal Component Analysis*, page 327352. MIT Press, Cambridge, MA, USA, 1999. ISBN 0262194163.

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.

Vin Silva and Joshua Tenenbaum. Unsupervised learning of curved manifolds. 01 2002. doi: 10.1007/978-0-387-21579-2_31.

Vin D. Silva and Joshua B. Tenenbaum. Global versus local methods in nonlinear dimensionality reduction. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems 15*, pages 721–728. MIT Press, 2003. URL http://papers.nips.cc/paper/2141-global-versus-local-methods-in-nonlinear-dimensionality-reduction.pdf.

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.

C. O. S. Sorzano, J. Vargas, and A. Pascual Montano. A survey of dimensionality reduction techniques, 2014.

J. S. Stanley III, G. Wolf, and S. Krishnaswamy. Manifold alignment with feature correspondence. *CoRR*, abs/1810.00386, 2018. URL http://arxiv.org/abs/1810.00386.

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.

Ming Tu, Jing Huang, Xiaodong He, and Bowen Zhou. Multiple instance learning with graph neural networks, 06 2019.

D. Tuia and G. Camps-Valls. Kernel manifold alignment. 04 2015.

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

Michail Vlachos, Carlotta Domeniconi, Dimitrios Gunopulos, George Kollios, and Nick Koudas. Non-linear dimensionality reduction techniques for classification and visualization. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD 02, page 645651, New York, NY, USA, 2002. Association for Computing Machinery. ISBN 158113567X. doi: 10.1145/775047.775143. URL https:

//doi.org/10.1145/775047.775143.

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.

C. Wang and S. Mahadevan. Multiscale manifold alignment. 09 2010.

C. Wang and S. Mahadevan. Heterogeneous domain adaptation using manifold alignment. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Two*, IJCAI'11, pages 1541–1546. AAAI Press, 2011. ISBN 978-1-57735-514-4. doi: 10.5591/978-1-57735-516-8/IJCAI11-259. URL http://dx.doi.org/10.5591/978-1-57735-516-8/IJCAI11-259.

Quan Wang. Kernel principal component analysis and its applications in face recognition and active shape models. *CoRR*, abs/1207.3538, 2012. URL http://arxiv.org/abs/1207.3538.

S. Wang, W. Chen, S.M. Xie, G. Azzari, and D.B. Lobell. Weakly supervised deep learning for segmentation of remote sensing imagery. *Remote Sens.*, 12:207, 2020.

Yong Wang, Jian-Bin Xie, and Yi Wu. Orthogonal discriminant analysis revisited. *Pattern Recognition Letters*, 84:149 – 155, 2016. ISSN 0167-8655. doi: https://doi.org/10.1016/j.patrec.2016.09.010.

Andrew Webb. A kernel approach to metric multidimensional scaling. pages 613–629, 02 2002.

Andrew R Webb and David Lowe. The optimised internal representation of multilayer classifier networks performs nonlinear discriminant analysis. *Neural Networks*, 3(4):367 – 375, 1990. ISSN 0893-6080. doi: https://doi.org/10.1016/0893-6080(90)90019-H.

Xiu-Shen Wei and Zhi-Hua Zhou. An empirical study on image bag generators for multi-instance learning. *Mach. Learn.*, 105(2):155198, November 2016. ISSN 0885-6125. doi: 10.1007/s10994-016-5560-1. URL https://doi.org/10.1007/s10994-016-5560-1.

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

Daniela M. Witten and Robert Tibshirani. Supervised multidimensional scaling for visualization, classification, and bipartite ranking. *Computational Statistics and Data Analysis*, 55(1):789 – 801, 2011. ISSN 0167-9473.

Hui Wu. *Weakly supervised learning on image manifolds*. PhD thesis, University of North Carolina at Charlotte, Charlotte, NC, 2015.

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. Xiao, B. Liu, and Z. Hao. A sphere-description-based approach for multiple-instance learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(2):242–257, Feb 2017. ISSN 0162-8828. doi: 10.1109/TPAMI.2016.2539952.

C. Xu, D. Tao, and Y. Rui. Large-margin weakly supervised dimensionality reduction. *31st International Conference on Machine Learning, ICML 2014*, 3:2472–2482, 01 2014.

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

S. Yan, D. Xu, B. Zhang, H. Zhang, Q. Yang, and S. Lin. Graph embedding and extensions: A general framework for dimensionality reduction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(1):40–51, Jan 2007.

Yixin Chen, Jinbo Bi, and J. Z. Wang. Miles: Multiple-instance learning via embedded instance selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12): 1931–1947, Dec 2006.

S. E. Yuksel, J. Bolton, and P. D. Gader. Landmine detection with multiple instance hidden markov models. In *2012 IEEE International Workshop on Machine Learning for Signal Processing*, pages 1–6, Sep. 2012. doi: 10.1109/MLSP.2012.6349734.

S. E. Yuksel, J. Bolton, and P. Gader. Multiple-instance hidden markov models with applications to landmine detection. *IEEE Transactions on Geoscience and Remote Sensing*, 53(12): 6766–6775, Dec 2015. ISSN 1558-0644. doi: 10.1109/TGRS.2015.2447576.

Z. Zhang, H. Zha, and M. Zhang. Spectral methods for semi-supervised manifold learning. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–6, June 2008. doi: 10.1109/CVPR.2008.4587381.

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

A. Zare, M. Cook, B. Alvey, and D. K. Ho. Multiple instance dictionary learning for subsurface object detection using handheld emi. In *Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XX, 94540G*, Proc. SPIE, May 2015. doi: 10.1117/12.2179177. URL https://doi.org/10.1117/12.2179177.

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.

Cha Zhang, John C. Platt, and Paul A. Viola. Multiple instance boosting for object detection. In Y. Weiss, B. Schölkopf, and J. C. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 1417–1424. MIT Press, 2006.

Q. Zhang and S. A. Goldman. Em-dd: An improved multiple-instance learning technique. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, pages 1073–1080. MIT Press, 2002.

Y. Zhang, X. Zheng, G. Liu, X. Sun, H. Wang, and K. Fu. Semi-supervised manifold learning based multigraph fusion for high-resolution remote sensing image classification. *IEEE Geoscience and Remote Sensing Letters*, 11(2):464–468, Feb 2014. ISSN 1545-598X. doi: 10.1109/LGRS.2013.2267091.

Yan Zhang, Zhao Zhang, Jie Qin, Li Zhang, Bing Li, and Fanzhang Li. Semi-supervised local multi-manifold isomap by linear embedding for feature extraction. *Pattern Recognition*, 76: 662 – 678, 2018. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2017.09.043. URL http://www.sciencedirect.com/science/article/pii/S0031320317303977.

Zhi-Hua Zhou and Jun-Ming Xu. On the relation between multi-instance learning and semi-supervised learning. In *Proceedings of the 24th International Conference on Machine Learning*, ICML 07, page 11671174, New York, NY, USA, 2007. Association for Computing Machinery. ISBN 9781595937933. doi: 10.1145/1273496.1273643. URL https://doi.org/10.1145/1273496.1273643.

Zhi-Hua Zhou and Min-Ling Zhang. Ensembles of multi-instance learners. In Nada Lavrač, Dragan Gamberger, Hendrik Blockeel, and Ljupčo Todorovski, editors, *Machine Learning: ECML 2003*, pages 492–502, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.

Zhi-Hua Zhou, Yu-Yin Sun, and Yu-Feng Li. Multi-instance learning by treating instances as non-i.i.d. samples. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML 09, page 12491256, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605585161. doi: 10.1145/1553374.1553534. URL https://doi.org/10.1145/1553374.1553534.

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.