Seminar Report

Applying Semi-Supervised Locally Linear Embedding

Department of Statistics
Ludwig-Maximilians-Universität München

By Lisa Wimmer
Under the supervision of Jann Goschenhofer, Ph.D.

Munich, month day$^{\text{th}}$, 2021

Abstract

Storyline

- Goal: present SS-LLE as a local, graph-based manifold learning method incorporating prior knowledge
- Step 0: define basic mathematical concepts required to understand argumentation (plus notation)
- Step 1: introduce idea of **isometry** (most basic: MDS)
- Step 2: introduce idea of **graph-based** models
  - Achieve non-linearity
  - Common structure: build graph → derive matrix as quadratic form over graph function → derive embedding from eigenvalue problem
  - Most basic: ISOMAP (global, dense, convex)
- Step 3: introduce idea of **locality**
  - Relax global to local isometry
  - Find sparse rather than dense matrices
  - **Laplacian eigenmaps** as concept in which the others can be generalized
    - Define weighting scheme for neighborhood
    - Use Laplacian to derive matrix
    - Solve sparse eigenvalue problem
- Step 4: introduce **local linearity**
  - **LLE**
    - Obtain weights via linear reconstructions
    - Can be shown to approximate graph Laplacian (Belkin & Niyogi (2006))
  - **Hessian LLE**
    - Replace Laplacian by Hessian
- Step 5: introduce **prior knowledge**
  - **SS-LLE**
  - Improve results by pre-specifying some manifold coordinates

Extended Abstract

CHECK AGAINST INTRODUCTION

The goal of this report is to lay out the theoretical framework behind the manifold learning technique of *semi-supervised locally linear embedding (SS-LLE)*, as proposed by Yang et al. (2006), and to put it to implementation for data sampled from manifolds.

Manifold learning in general is concerned with dimensionality reduction. As data analysis employs increasingly high-dimensional data, it is frequently necessary to scale down the number of features to ensure models work as desired and remain interpretable. Dimensionality reduction is justified by the assumption that data observed in D dimensions often truly lie on a d-dimensional manifold (d-manifold), i.e., the d-dimensional generalization of a curved surface, embedded in $\mathbb{R}^D$ (with $d \ll D$). As an example for this phenomenon one might consider image data showing objects in different poses. While images are typically stored in high-dimensional pixel representations, intuitively, it is in fact a very small number of features causing the variation in the data.

A crucial property of d-manifolds embedded in $\mathbb{R}^D$ is their local topological equivalence to $\mathbb{R}^d$. This locally Euclidean behavior is exemplified by a sphere embedded in $\mathbb{R}^3$: although the sphere as a whole is entirely non-linear, on sufficiently small patches of its surface it behaves just like a flat plane in $\mathbb{R}^2$. It is precisely this fact that allows manifold coordinates to be mapped to $\mathbb{R}^d$ in a reduction of dimensionality. The goal is now to learn this mapping in an unsupervised manner. Mapping manifold coordinates to $\mathbb{R}^d$ is in general not equivalent to simple projection onto the d-dimensional coordinate hyperplanes. Instead, models must learn the intrinsic neighborhood structure of the manifold to establish a notion of "nearness" between points. As the sphere example demonstrates, standard distance metrics do not apply (globally) since points on general manifolds are connected by curved paths rather than straight lines.

Some manifold learning techniques try to retain global isometry by mapping pairwise distances to $\mathbb{R}^d$. For instance, *multi-dimensional scaling (MDS)* does so using Euclidean distances, thereby limiting the manifolds it can learn to linear ones, while *ISOMAP* generalizes this approach to non-linear manifolds by applying geodesic distances. Research indicates, however, that for non-convex manifolds it is more effective to preserve local structures only. Otherwise, solutions are prone to shortcuts, i.e., placing points close in $\mathbb{R}^D$ next to each other when they lie in fact on quite different parts of the manifold. In order to avoid such miscalculations, sparse techniques focus on merely local neighborhood structures, modeled through weighted graph representations. The information from these graphs is then condensed into a sparse matrix. Eventually, the principal eigenvectors of this matrix yield the desired low-dimensional coordinates.

One such local graph-based technique is *Laplacian eigenmaps*, a method in whose general framework other techniques may be interpreted. It employs the graph Laplacian and does well in preserving locality, yet is less adept at determining local linearity. This shortcoming is mitigated by *locally linear embedding (LLE)*and its variants. LLE is based on the idea that the embedded manifold may be approximated by locally linear neighborhoods in $\mathbb{R}^D$. Since weights resulting from linear reconstruction are believed to reflect the intrinsic geometry of the manifold, they are topological

properties and as such invariant to rotations, rescalings, and translations. By consequence, these same weights should also reconstruct points in d dimensions. LLE thus maps vicinity structures to the d-dimensional subspace and finds the coordinates that preserve them best. This requires solving the least-squares problem of minimizing reconstruction error and then the sparse eigenvalue problem of minimizing embedding cost. Convexity of both sub-problems ensures globality of local optima. A later proposition, *Hessian LLE (H-LLE)*, may be viewed as an algorithmic variant of LLE and a conceptual variant of Laplacian eigenmaps using the Hessian en lieu of the Laplacian.
These approaches have been shown to successfully retrieve manifold structures in different applications. However, their fully unsupervised functionality offers a drawback: they may fail to find a low-dimensional embedding that has an actual reflection in the real-life setting. Such situations might require the specification of some prelabeled instances. Also, it may simply be the case that manual analysis of a subset of the data is available at low cost.
When prior knowledge is at hand it is only natural to use it. Therefore, Yang et al. (2006) proposed SS-LLE as an extension to LLE that is able to harvest prior information. Both exact and inexact knowledge, the latter regularized with an uncertainty coefficient, are applicable. The information is incorporated in the second step of the algorithm by fixing some of the sought-for coordinates in advance. Perhaps unsurprisingly, Yang et al. (2006) find that careful selection of the prior points to be maximally scattered across the manifold surface works better than random sampling. Indeed, the presented results indicate considerable success of their technique.
It is the aim of this report to (1) reproduce these results, thereby creating an open-source implementation of SS-LLE, and (2) to apply SS-LLE to further manifold learning tasks for a more thorough assessment of its performance.

Contents

List of Symbols

$D \in \mathbb{N}$	Number of observed dimensions
$d \in \mathbb{N}$	Number of dimensions of embedded manifold
$m \in \mathbb{N}$	Number of dimensions of low-dimensional representation
$N \in \mathbb{N}$	Number of observed data points
$\mathcal{M} \subset \mathbb{R}^D$	d-manifold embedded in $\mathbb{R}^D$
$\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_N) \in (\mathbb{R}^D)^N$	Observed coordinates of data sampled from $\mathcal{M}$
$\boldsymbol{Y} = (\boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_N) \in (\mathbb{R}^m)^N$	Learned coordinates of low-dimensional representation of data

List of Figures

List of Tables

1 Introduction

Machine learning problems increasingly employ data of high dimensionality. While a large amount of samples is beneficial to learning, high-dimensional feature spaces, such as in speech recognition or gene processing, pose serious obstacles to the performance and convergence of most algorithms (Cayton, 2005).
Three aspects strike as particularly problematic: computational operations, interpretation of results, and geometric idiosyncrasies. Computational cost must be considered but is becoming less of an issue with technological evolution (Leist et al., 2009). By contrast, the demand for explainable results (for reasons of, say, safety or ethics) is rather intensified by the advance of complex technology. Alas, interpretation in more than a few dimensions is virtually inaccessible to humans (Doshi-Velez and Kim, 2017). The geometric aspect is often addressed as *curse of dimensionality*, a term subsuming various phenomena of high-dimensional spaces. It is generally not straightforward to infer properties of objects in these spaces as geometric intuition developed in lower dimensions can be misleading. Crucially, the exponential increase of spatial volume induces sparsity. Consequences of this behavior are, among others, a sharp incline in the number of points required to sample the feature space and a loss in meaningfulness of distances. Many learners, however, rely on these concepts[1] and see their functionality deteriorate (Verleysen and Francois, 2005).
These challenges make the case for *dimensionality reduction*, that is, the endeavor of compressing problem dimensionality to a manageable size. Far from undue simplification, dimensionality reduction is justified by the belief that the latent data-generating process is indeed of much lower dimension than is observed. Consider, for example, image data showing objects in different poses. Such data are typically stored in high-dimensional pixel representations, yet it is reasonable to suppose that variation in these images is in fact caused by a small number of latent features. More formally, the data are assumed to lie on a d-dimensional *manifold* embedded in the D-dimensional observation space, with $d \ll D$ (Cayton, 2005).
A crucial property of d-manifolds, i.e., the d-dimensional generalization of a curved surface, embedded in $\mathbb{R}^D$, is their local topological equivalence to $\mathbb{R}^d$ (Ma and Fu, 2011). It is precisely this fact that allows manifold coordinates to be mapped to $\mathbb{R}^d$ in a reduction of dimensionality[2]. The goal is now to learn this mapping in an unsupervised manner (Cayton, 2005). Mapping manifold coordinates to $\mathbb{R}^d$ is in general not equivalent to simple projection onto the d-dimensional coordinate hyperplanes. Instead, models must learn the intrinsic neighborhood structure of the manifold to establish a notion of "nearness" between points. Standard distance metrics do not apply (globally) as points on general manifolds are connected by curved paths rather than straight lines (Ma and Fu, 2011).
Various approaches have been proposed to retrieve points' manifold coordinates. A taxonomy may, for example, be found in van der Maaten et al. (2009). Many rely on spectral techniques, trying to find a matrix representation of the data whose principal eigenvectors are used to span a d-dimensional subspace. One group of spectral methods attempts to retain global isometry by mapping pairwise distances

[1]For instance, consider support vector machines and k-nearest neighbors, both of which rely on distances, or tuning, which often requires extensive sampling of the hyperparameter space.
[2]The most intuitive example of this is probably the representation of the Earth, which is a two-dimensional manifold enclosed in three-dimensional space, on two-dimensional maps.

to $\mathbb{R}^d$. Among them, some are based on Euclidean distances and thus confined to learning linear embeddings (such as *principal component analysis (PCA)* or *multi-dimensional scaling (MDS)*). Since linearity is a strong assumption that will not hold for general manifolds, non-linear techniques are more widely applicable (van der Maaten et al., 2009). For example, *ISOMAP* achieves non-linearity by applying geodesic distances in the MDS setup (Tenenbaum et al., 2000). Research indicates, however, that for non-convex manifolds it is more effective to preserve local structures only. Otherwise, solutions are prone to shortcuts, i.e., placing points close in $\mathbb{R}^D$ next to each other when they lie in fact on quite different parts of the manifold (Belkin and Niyogi, 2003). In order to avoid such miscalculations, sparse techniques focus on merely local neighborhood structures, modeled through weighted graph representations. The information from these graphs is then condensed into a sparse matrix. Eventually, the principal eigenvectors of this matrix yield the desired low-dimensional coordinates (van der Maaten et al., 2009).
One such local graph-based technique is *locally linear embedding (LLE)*, the unsupervised algorithm SS-LLE builds upon (Roweis and Saul, 2000). LLE is based on the idea that the embedded manifold may be approximated by locally linear neighborhoods in $\mathbb{R}^D$. Weights for the resulting graph are obtained by linear reconstruction of points from their neighbors. As these weights are believed to reflect the intrinsic geometry of the manifold, they are topological properties and should as such also reconstruct points in d dimensions. LLE thus maps vicinity structures to the d-dimensional subspace and finds the Euclidean coordinates that preserve them best by means of spectral decomposition (Roweis and Saul, 2000).
Much of the theoretical foundation for LLE has been discussed only in later work. In particular, Belkin and Niyogi (2001) proposed *Laplacian eigenmaps*, a method which employs the graph Laplacian, and provided evidence for the fact that, under certain assumptions, LLE may be generalized to the same framework (Belkin and Niyogi, 2003). A later proposition by Donoho and Grimes (2003), *Hessian LLE (H-LLE)*, may be viewed as an algorithmic variant of LLE and a conceptual variant of Laplacian eigenmaps (using the Hessian en lieu of the Laplacian). The theoretical link between LLE and Laplacian eigenmaps, centered around the Laplace-Beltrami operator, has recently been found to hold less generally than previously assumed (Wu and Wu, 2018). It still appears beneficial to interpret all methods in this common framework also found by Bengio et al. (2003); a more thorough study of convergence guarantees is left to future research.
The above approaches have been shown to successfully retrieve manifold structures in different applications (Wu and Wu, 2018). However, their fully unsupervised functionality offers a drawback: they may fail to find a low-dimensional embedding that has an actual reflection in the real-life setting. Such situations might require the specification of some pre-labeled instances. Also, it may simply be the case that manual analysis of a subset of the data is available at low cost (Yang et al., 2006). When prior knowledge is at hand it is only natural to use it. Therefore, Yang et al. (2006) proposed *semi-supervised locally linear embedding (SS-LLE)*, an extension to LLE that is able to harvest prior information.

Indeed, the presented results indicate considerable success of their technique. It is the aim of this report to (1) reproduce these results, thereby creating an open-source implementation of SS-LLE, and (2) to apply SS-LLE to further manifold learning tasks for a more thorough assessment of its performance. The rest of the report is organized as follows: chapter 2 provides a mathematical framework where fundamental concepts are briefly introduced; chapter 3 explains the framework of local graph-based manifold learning; chapter 4 presents SS-LLE in detail; chapter 5 discusses the results of the conducted experiments; and chapter 6 draws final conclusions.

2 Basic Manifold Theory

2.1 Concepts in Manifold Learning

This chapter introduces the main geometric concepts considered necessary to provide a solid understanding of SS-LLE[3]. It must be noted that everything discussed here is presented through the lens of machine learning, deliberately forsaking the generality inherent to topology. Therefore, assuming features can be represented by coordinates in D-dimensional Euclidean space, all concepts are examined with regard to their meaning in $\mathbb{R}^D$. Dimensionality reduction techniques take the data observed in $\mathbb{R}^D$ to actually lie in a d-dimensional topological space that is not necessarily Euclidean but exhibits some specific properties.

Topological spaces. A *topological space* is constituted by a set T equipped with a *topology* $\mathcal{T}$. A topology is a general way of describing relations between elements in T. Consider a function $\mathcal{T} : T \rightarrow 2^T, t \mapsto \mathcal{T}(t)$, which assigns to $t \in T$ a set of subsets of T called *neighborhoods*. For $\mathcal{T}$ to be a topology[4] on T, the following properties must hold (Brown, 2006):

1. If $\mathcal{T}$ is a neighborhood of t, then $t \in \mathcal{T}$.
2. If $\mathcal{T}$ is a subset of T containing a neighborhood of t, then $\mathcal{T}$ is a neighborhood of t.
3. The intersection of two neighborhoods of t is again a neighborhood of t.
4. Any neighborhood $\mathcal{T}$ of t contains a neighborhood $\mathcal{T}'$ of t such that $\mathcal{T}$ is a neighborhood of each element in $\mathcal{T}'$.

Note that, in this general definition, neighborhoods are based on an abstract notion of "nearness". Learning the structure of a topological space effectively boils down to learning neighborhood relations. In Euclidean topological space these are directly based on distance: neighborhoods around t are constructed by ϵ-balls containing all elements within a Euclidean distance of ϵ from t. The resulting topology is also called the *metric topology* (McCleary, 2006).
Topological spaces in general are not accessible via distances (or angles, for that matter) known from Euclidean spaces. The ultimate goal therefore is the interpretation of the data in a space that is again Euclidean, albeit of lower dimensionality, where such concepts are meaningful. The next step is then to study how (potentially highly non-linear) topological spaces relate to $\mathbb{R}^d$.

[3] Obviously, the list of concepts discussed is by no means extensive. Theory is presented much more in detail (and mathematical rigor) in, for example, good book.

[4] Alternative definitions employ open subsets of T, see for example Waldmann (2014).

Homeomorphisms. Consider two topological spaces $(S, \mathcal{T}_S)$, $(T, \mathcal{T}_T)$ (denoted by the respective shorthands S, T from here) and a mapping function $f : S \rightarrow T$. If f is bijective and continuous and $f^{-1} : T \rightarrow S$ is also continuous, f is called a *homeomorphism* (Brown, 2006). Topological spaces for which such a mapping exists are *homeomorphic* to each other. Any properties of S that T shares when it is homeomorphic to S are referred to as topological properties. Two homeomorphic spaces are thus topologically equivalent (McCleary, 2006).
If there exists a non-negative integer d such that for every s in a topological space S a local neighborhood $U \ni s$, $U \subset S$, is homeomorphic to an open subset of $\mathbb{R}^d$, S is *locally Euclidean*[5] (Ma and Fu, 2011). In other words, there is a homeomorphism $f : U \rightarrow \mathbb{R}^d$ for every element in S. The neighborhoods U are also referred to as *coordinate patches* and the associated maps f are called *coordinate charts* (Cayton, 2005). In local neighborhoods S then behaves like $\mathbb{R}^d$ (Ma and Fu, 2011).

Manifolds. *Manifolds* are now precisely such locally Euclidean topological spaces, with some additional properties. For a topological space $\mathcal{M}$ to be a d-dimensional manifold[6] (also: d-manifold) it must meet the following conditions (Waldmann, 2014):

1. $\mathcal{M}$ is Hausdorff.
2. $\mathcal{M}$ is second-countable.
3. $\mathcal{M}$ is locally homeomorphic to $\mathbb{R}^d$.

The Hausdorff condition is a separation property and ensures that for any two distinct points from $\mathcal{M}$ disjoint neighborhoods can be found (Brown, 2006). Second-countability restricts the manifold's size via the number of open sets it may possess (Waldmann, 2014).

Embeddings. Recall that the data are observed in $\mathbb{R}^D$ but taken to lie on $\mathcal{M}$, locally homeomorphic to $\mathbb{R}^d$. This implies the assumption $\mathcal{M} \subset \mathbb{R}^D$ and $\mathcal{M}$ is said to be *embedded* in the ambient D-dimensional Euclidean space (Cayton, 2005). The associated *embedding* is but a map $f : \mathcal{M} \rightarrow \mathbb{R}^D$ whose restriction to $\mathcal{M}$ is a homeomorphism (Brown, 2006), or, more specifically, the canonical inclusion map identifying points on the manifold as particular points of $\mathbb{R}^D$ (Waldmann, 2014). It can be shown that $K = 2d + 1$ is sufficient to create an embedding (Ma and Fu, 2011). Figure 1 shows the so-called *S-curve* manifold embedded in $\mathbb{R}^3$. Clearly, the S-curve as a whole is far from linear, but locally homeomorphic to $\mathbb{R}^2$ and thus intrinsically two-dimensional.

Figure 1: 10,000 points sampled from the S-curve manifold. *Source:* own representation, inspired by implementation in `Python`'s `scikit-learn` library (Pedregosa et al., 2011).

[5] For locally Euclidean topological spaces it is thus meaningful to speak of elements as points.
[6] $\mathcal{M}$ is again a shorthand, omitting the explicit notation of the corresponding topology.

Geodesics. One last aspect remains open and shall be briefly touched here, namely how to handle distances on manifolds where Euclidean metrics are not meaningful. Rather than measuring "shortcuts" between points across $\mathbb{R}^D$ (where, for instance, points in the red upper area of figure 1 would be considered deceptively close to points in the cyan mid area), it makes intuitive sense to constrain distances to the manifold surface. In order to enable the construction of such a metric, manifolds must fulfill two additional properties: *smoothness*[7] and *connectedness*[8] (Ma and Fu, 2011). For smooth, connected manifolds, *geodesic distance* is the length of the shortest curve (*geodesic*) on $\mathcal{M}$ between two points on $\mathcal{M}$ as measured by arc-length[9]. Intuitively, geodesic distance can be identified with Euclidean distance in Euclidean spaces where shortest curves are just straight lines (Ma and Fu, 2011).

2.2 Formal Goal of Manifold Learning

Building on the above concepts, the data situation in manifold learning might be summarized as follows: data are observed in $\mathbb{R}^D$ but assumed to be really samples from a d-manifold $\mathcal{M}$ embedded in $\mathbb{R}^D$, meaning they can be analyzed in $\mathbb{R}^d$ if a faithful translation between respective coordinates in $\mathbb{R}^D$ and $\mathbb{R}^d$ is found. The challenge is thus to unravel the manifold in a way that preserves its intrinsic structure to maximum extent (Saul et al., 2006). In the case of figure 1, this may be imagined as "flattening out" the S-curve. This goal shall be formalized in a way that will be referenced throughout the remainder of this report and that is inspired by the works of Cayton (2005) and Saul et al. (2006).

Given. Data $\mathcal{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_N)$, with $\boldsymbol{x}_i \in \mathbb{R}^D \ \forall i \in \{1, 2, ..., N\}$ and $N, D \in \mathbb{N}$. $\mathcal{X}$ thus consists of N real-valued data vectors observed in D dimensions. The true data-generating process is taken to have dimensionality $d \in \mathbb{N}$, such that $\mathcal{X}$ is in fact a sample from a smooth, connected d-manifold with $\mathcal{X} \sim \mathcal{M} \subset \mathbb{R}^D$. $\mathcal{M}$ may be described by a single coordinate chart[10] $\psi : \mathcal{M} \rightarrow \mathbb{R}^d$. For manifold learning methods to yield satisfying results, $\mathcal{M}$ is always assumed to be sampled well by $\mathcal{X}$.

Goal. Find the d-dimensional representation of the data, i.e., compute $\mathcal{Y} = (\boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_N)$, where $\boldsymbol{y}_i = \psi(\boldsymbol{x}_i) \in \mathbb{R}^d \ \forall i \in \{1, 2, ..., N\}$. The map ψ itself is not always explicitly retrieved.

Note that, while D is given a priori, the intrinsic dimensionality d is often unknown in real-life applications. Due to the lack of (finite-sample) convergence guarantees for many methods dimensionality estimations may not always be equal to d. $\mathcal{Y}$ as found by manifold learning techniques must therefore be expected to differ from the true coordinates and, in particular, to even have incorrect dimension (Saul et al., 2006). Notwithstanding this potential gap, solutions of the subsequently presented

[7] The smoothness property is based on differentiability of coordinate charts and ensures that concepts of curvature, length and angle remain meaningful (Ma and Fu, 2011). A detailed derivation may be found, for example, in Mukherjee (2015).

[8] Connectedness means that no separation $\{U, V\}$ of a manifold $\mathcal{M}$ exists with open, non-empty and disjoint $U, V \subset \mathcal{M}$, $\mathcal{M} = U \cup V$. This may be loosely put as paths linking arbitrary pairs of manifold points (McCleary, 2006).

[9] Geodesics are but a peripheral note here; for a precise definition see Ma and Fu (2011).

[10] This is possible for any smooth, compact manifold (Cayton, 2005).

methods will be denoted by $\mathcal{Y} \in (\mathbb{R}^d)^N$ to avoid overly complicated notation.

3 Local Graph-Based Manifold Learning

3.1 Principles of Local Graph-Based Manifold Learning

3.1.1 General Concept

After the goal of manifold learning has been formalized, it shall now be laid out how LLE, as the conceptual parent of SS-LLE, approaches the problem. In order to provide some background, this chapter will first sketch the fundamental idea of local graph-based techniques. Methods subsumed under this term arise from a variety of geometric intuitions and computational implementations. Interestingly, they still share a common structure that allows for interpretation in a framework applicable to general spectral methods (Bengio et al., 2003).
It must be noted that, while the algorithmic similarity is obvious, it is less clear whether the established theoretical connection holds as well. Belkin and Niyogi (2001) derived their argumentation for using the Laplacian in combination with the heat kernel for Laplacian eigenmaps from the analogy of the graph Laplacian and the Laplace-Beltrami operator on manifolds. In subsequent studies they confirmed the legitimacy of this approach by providing proofs of convergence (Belkin and Niyogi, 2008). It is widely assumed that LLE may also be cast into this framework by considering the eigenvalue problem it solves as an approximation to the graph Laplacian (Belkin and Niyogi (2003), Donoho and Grimes (2003)). H-LLE neatly fits the same idea as it bears strong algorithmic resemblance to LLE, hence the name, and can be viewed as a conceptual variant of Laplacian eigenmaps, substituting the Laplacian for the Hessian (Donoho and Grimes, 2003). Recent studies, however, indicate that the asymptotic convergence of LLE to the Laplace-Beltrami operator might be subject to additional regularization assumptions (Wu and Wu, 2018). The report therefore abstains from a direct translation of both LLE and H-LLE to variations of Laplacian eigenmaps. Still, the three methods have a triangular relationship at least, where H-LLE borrows from the theory behind Laplacian eigenmaps and employs an algorithm strongly reminiscent of LLE (Donoho and Grimes, 2003). For the sake of a more integrated picture, the following explanations will therefore be made in the shared framework, with an emphasis on the algorithmic commonalities.

To begin with, all spectral methods of manifold learning perform two central tasks: first, find a matrix representation for the D-dimensional data; second, perform a spectral decomposition of this matrix such that the data can be expressed via coordinates on the principal (top or bottom) eigenvectors (Saul et al., 2006). Precisely how the matrix is constructed determines the kind of intrinsic structure that can be learned and preserved (Cayton, 2005). Local-graph based methods find theirs in a way that allows for two general desiderata in manifold learning: non-linearity of detectable manifolds, and locality of preserved structures.

Non-linearity. Early methods of manifold learning attempted to preserve linear structures observed in $\mathbb{R}^D$. The two most prominent of these are probably *principal component analysis (PCA)* and *multi-dimensional scaling (MDS)*. Both eventually

arrive at the same result; while PCA retains the global covariance structure, decomposing the covariance matrix, MDS keeps global pairwise Euclidean distances by decomposition of the Gram matrix (Cayton, 2005). The central drawback of this approach is the confinement to learning linear mappings. Graph-based methods acknowledge the fact that, in general, data will rather lie on non-linear manifolds. The techniques studied here define functionals over smooth mappings from the manifold to Euclidean space and use these functionals to find optimal embeddings.
They model the intrinsic structure through neighborhood relations, mapping points that lie close on the manifold surface to nearby locations. Neighborhoods are characterized via weighted graphs. In effect, these graphs are a discretized approximation of the underlying manifold $\mathcal{M}$, still assuming $\mathcal{M}$ is sampled well by $\mathcal{X}$ (Saul et al., 2006). They are used to find a numerical matrix representation of the data by approximating certain functionals on the manifold. *ISOMAP* is one of the first graph-based techniques that extends the approach of MDS to geodesic distances, meaning “nearness” between points is expressed by distance on the manifold surface rather than in the ambient Euclidean space (Tenenbaum et al., 2000).

Local isometry. ISOMAP relies on a central assumption[11] that turns out to be too restrictive for many settings, namely global isometry: for an arbitrary pair of points on $\mathcal{M}$, their geodesic distance is preserved in the mapping to $\mathbb{R}^d$ (Donoho and Grimes, 2003). This assumption is violated when $\mathcal{M}$ is geodesically non-convex, i.e., it is not isometric to a convex subset of Euclidean space[12] (Saul et al., 2006). Local graph-based methods relax the isometry assumption to a local one. Rather than for arbitrary pairs of points, it need only hold for neighboring ones (Donoho and Grimes, 2003). Where global solutions are sometimes too coarse, local methods thus allow for tracing non-convex behavior. In doing so, they produce sparse matrix representations (Cayton, 2005).

Laplacian eigenmaps, LLE and H-LLE all belong to the family of local graph-based techniques whose functionality may be summarized as follows:

1. Compute neighborhoods of input data.
2. Construct a sparse weighted graph from these neighborhood relations.
3. Condense the graph information in a matrix.
4. Learn an embedding from the eigenvectors of this matrix.

Crucially, the complex manifold learning problem is decomposed into a sequence of tractable optimization steps (Saul et al., 2006). The subsequent chapters will explain these steps in some more detail and afterwards lay out how they are operationalized in Laplacian eigenmaps, LLE and H-LLE.

3.1.2 Neighborhood Graphs

Approximating the intrinsic structure of $\mathcal{M}$ by a graph representation requires the determination of a *neighborhood* around point. A neighborhood of $\boldsymbol{x} \in \mathcal{X}$ is but a subset of $\mathcal{X}$ containing another, open subset of $\mathcal{X}$ of which $\mathcal{X}$ is an element. Members

[11] In fact, there is a second assumption of convex parameter spaces that is omitted here, see Tenenbaum et al. (2000).

[12] Intuitively, this can be imagined as the manifold containing “holes”.

of the neighborhood are called neighbors of $\mathcal{X}$. In metric spaces neighborhoods are defined via distances and therefore translate to open balls around each point (Waldmann, 2014). This distance-based construction now locally applies to manifolds as a direct consequence of their local isometry to the Euclidean observation space (Ma and Fu, 2011). There are two principal ways to build a neighborhood around $\boldsymbol{x} \in \mathcal{X}$, both of which usually employ squared Euclidean distances[13], denoted by $\| \cdot \|^2$. Let $\mathcal{N} : \mathcal{X} \rightarrow \mathcal{X}^\ell, \boldsymbol{x} \mapsto \mathcal{N}(\boldsymbol{x})$ be a constructor that assigns a set of neighbors to $\boldsymbol{x}$. The first possibility is to restrict the size of the neighborhood to the k points with the smallest distance to $\boldsymbol{x}$, such that $\ell = k$ and $\mathcal{N}_k(\boldsymbol{x}) = \{\boldsymbol{x}_j \in \mathcal{X} : \|\boldsymbol{x} - \boldsymbol{x}_j\|^2 \leq \gamma\}$, with $\gamma \in \mathbb{R}$ being the k-th instance of ordered pairwise distances. Alternatively, the neighborhood may be constructed by collecting all points that have a maximum distance of $\epsilon \in \mathbb{R}$ to $\boldsymbol{x}$, yielding $\mathcal{N}_\epsilon(\boldsymbol{x}) = \{\boldsymbol{x}_j \in \mathcal{X} : \|\boldsymbol{x} - \boldsymbol{x}_j\|^2 \leq \epsilon\}$ and $\ell = |\mathcal{N}_\epsilon(\boldsymbol{x})|$ (He et al., 2005). Both k and ϵ are hyperparameters that must be specified up-front. The choice of neighborhood size reflects beliefs about the extent to which $\mathcal{M}$ is locally linear, smaller neighborhoods corresponding to a higher degree of non-linearity, and may affect strongly performance (Sudderth, 2002).

The thus defined neighborhoods can now be described by a *neighborhood graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where input points form vertices and edges indicate neighborhood relations (Belkin and Niyogi, 2001). Each vertex is connected to its k nearest neighbors or all points within ϵ-radius, depending on the neighborhood definition. It is easy to see that k-neighborhoods are an asymmetric notion; for one point to be among another's k nearest neighbors the reverse need not be true. Building upon k-neighborhoods therefore leads to directed graphs (He et al., 2005). An example for such a directed graph is given by figure 2, showing 2-neighborhoods for seven fictional data points. Conversely, the ϵ-distance boundary holds in both directions and produces undirected graphs (He et al., 2005).

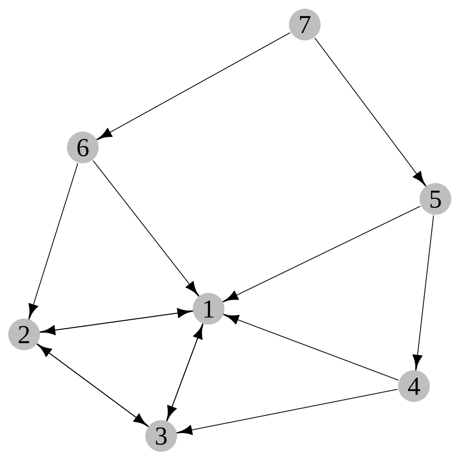

Figure 2: Exemplary neighborhood graph for seven fictional data points where outgoing arrows point to members of the vertex's respective k-neighborhood, $k = 2$. *Source:* own representation.

In order for neighborhood graphs[14] to translate into a low-dimensional embedding, the information they hold must be converted to a numerical representation. This is achieved by assigning weights to graph edges and constructing a matrix from these (Belkin and Niyogi, 2003). In fact, the

[13] In principle, alternative metrics are equally applicable, for instance such that measure angles (Belkin and Niyogi, 2004).

[14] Graphs are assumed to be connected, i.e., no vertex is left without at least one edge linking it to another vertex. k-neighborhoods tend to yield connected graphs more often than ϵ-neighborhoods. If the connectedness assumption does not hold globally, each connected subgraph must be considered separately (Belkin and Niyogi, 2001).

3.1.3 Matrix Representation of Neighborhood Graphs

3.1.4 Solving Eigenwertproblems

- Eigenvectors, eigenvalues
- Spectral decomposition

3.2 Techniques of Local Graph-Based Manifold Learning

3.2.1 Laplacian Eigenmaps

- Notion of locality
- Laplacian eigenmaps

3.2.2 Locally Linear Embedding (LLE)

- Notion of local linearity
- Approximation of graph Laplacian

3.2.3 Hessian Locally Linear Embedding (H-LLE)

- Hessian instead of Laplacian (eigenmaps)
- Hessian instead of LS fit (LLE)

4 Semi-Supervised Locally Linear Embedding (SS-LLE)

4.1 Employment of Prior Information

- Why use labels in the first place?
- How will that help?
- How do we even find prior points?
- Exact vs inexact knowledge

4.2 SS-LLE Algorithm

- What is different wrt standard LLE?

4.3 Strengths and Drawbacks of SS-LLE

Theoretical convergence? (e.g., ISOMAP has this)
Determination of d: actually requires to know d, right? Must be automatically known if prior points are known
Potential shortcoming: what if manifold is not well-sampled? Not a problem with synthetic data, but IRL. But probably problematic with all manifold approaches
Also: generalization to new points (w/o recomputing everything) neighborhood-preserving propositions

5 Experiment Results

5.1 Experimental Design

5.1.1 Software Implementation

5.1.2 Evaluation Framework

5.2 Replication of Original Results on the Incomplete Tire

5.2.1 Incomplete Tire Data

5.2.2 Choice of Hyperparameter and Prior Point Configuration

5.3 Application on Swiss Roll and Guglhupf Data

5.3.1 Swiss Roll and Guglhupf Data

5.3.2 Choice of Hyperparameter and Prior Point Configuration

5.4 Results and Discussion

6 Conclusion

Lorem ipsum

A Appendix

Lorem ipsum

B Electronic Appendix

Data, code and figures are provided in electronic form.

References

Belkin, M. and Niyogi, P. (2001). Laplacian eigenmaps and spectral technique for embedding and clustering, *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, p. 585–591.

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

Belkin, M. and Niyogi, P. (2004). Semi-supervised learning on riemannian manifolds, *Machine Learning* **56**: 209–239.

Belkin, M. and Niyogi, P. (2008). Towards a theoretical foundation for laplacian-based manifold methods, *Journal of Computer and System Sciences* **74**(8): 1289–1308.

Bengio, Y., Paiement, J.-F., Vincent, P., Delalleau, O., Roux, N. L. and Ouimet, M. (2003). Out-of-sample extensions for lle, isomap, mds, eigenmaps, and spectral clustering, *Proceedings of the 16th International Conference on Neural Information Processing Systems*, MIT Press, p. 177–184.

Brown, R. (2006). *Topology and Groupoids. A Geometric Account of General Topology, Homotopy Types and the Fundamental Groupoid*, 2 edn, Createspace.

Cayton, L. (2005). Algorithms for manifold learning, *Technical Report CS2008-0923*, University of California, San Diego (UCSD).

Donoho, D. L. and Grimes, C. (2003). Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data, *Proceedings of the National Academy of Sciences of the United States of America* **100**(10): 5591–5596.

Doshi-Velez, F. and Kim, B. (2017). Towards a rigorous science of interpretable machine learning, *arXiv: Machine Learning* .

He, X., Cai, D., Yan, S. and Zhang, H.-J. (2005). Neighborhood preserving embedding, *Proceedings of the Tenth IEEE International Conference on Computer Vision*.

Leist, A., Playne, D. P. and Hawick, K. A. (2009). Exploiting graphical processing units for data-parallel scientific applications, *Concurrency and Computation. Practice and Experience* **21**(18): 2400–2437.

Ma, Y. and Fu, Y. (2011). *Manifold Learning. Theory and Applications*, CRC Press.

McCleary, J. (2006). *A First Course in Topology. Continuity and Dimension*, American Mathematical Society.

Mukherjee, A. (2015). *Differential Topology*, 2 edn, Springer.

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

Roweis, S. T. and Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding, *Science* **290**(5500): 2323–2326.

Saul, L. K., Weinberger, K. Q., Sha, F., Ham, J. and Lee, D. D. (2006). Spectral methods for dimensionality reduction, *in* O. Chapelle, B. Scholkopf and A. Zien (eds), *Semi-Supervised Learning*, MIT Press Scholarship Online, chapter 1.

Sudderth, E. B. (2002). Nonlinear manifold learning part ii 6.454 summary.

Tenenbaum, J. B., de Silva, V. and Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction, *Science* **290**(5500): 2319–2322.

van der Maaten, L., Postma, E. and van den Herik, J. (2009). Dimensionality reduction: A comparative review, *Technical Report TiCC TR 2009-005*, Tilburg University.

Verleysen, M. and Francois, D. (2005). The curse of dimensionality in data mining and time series prediction, *in* J. Cabestany, A. Prieto and F. Sandoval (eds), *Computational Intelligence and Bioinspired Systems*, Springer.

Waldmann, S. (2014). *Topology. An Introduction*, Springer.

Wu, H.-T. and Wu, N. (2018). Think globally, fit locally under the manifold setup: Asymptotic analysis of locally linear embedding, *Ann. Stat* **246**(6B): 3805–3837.

Yang, X., Fu, H., Zha, H. and Barlow, J. (2006). Semi-supervised nonlinear dimensionality reduction, *Proceedings of the 23rd International Conference on Machine Learning*.

Declaration of Authorship

I hereby declare that the report submitted is my own unaided work. All direct or indirect sources used are acknowledged as references. I am aware that the Thesis in digital form can be examined for the use of unauthorized aid and in order to determine whether the report as a whole or parts incorporated in it may be deemed as plagiarism. For the comparison of my work with existing sources I agree that it shall be entered in a database where it shall also remain after examination, to enable comparison with future Theses submitted. Further rights of reproduction and usage, however, are not granted here. This paper was not previously presented to another examination board and has not been published.