Pattern Recognition Letters
journal homepage: www.elsevier.com

Manifold Matching using Shortest-Path Distance and Joint Neighborhood Selection

Cencheng Shen[a,b], Joshua T. Vogelstein[a,c], Carey E. Priebe[a,d,**]

[a]*Center for Imaging Science, Johns Hopkins University*
[b]*Department of Statistics, Temple University*
[c]*Department of Biomedical Engineering and Institute for Computational Medicine, Johns Hopkins University*
[d]*Department of Applied Mathematics and Statistics, Johns Hopkins University*

ABSTRACT

Exploring and matching datasets of multiple modalities has become an important task in data analysis. Most existing matching methods rely on embedding and transformation techniques for datasets of a single modality without fully utilizing the correspondence information, which often yields sub-optimal matching results. In this paper, we propose a new nonlinear manifold matching algorithm using shortest-path distance and joint neighborhood selection. Specifically, a joint graph is built for all modalities. Then the shortest-path distance within each modality is calculated from the joint neighborhood graph, followed by embedding into and matching in a common low-dimensional Euclidean space. Compared to existing popular algorithms, our approach exhibits superior performance for matching disparate datasets of multiple modalities.

1. Introduction

In today's world, it is becoming increasingly important to deal effectively with large amounts of high-dimensional data. For the purpose of data analysis, it is imperative to consider dimension reduction and embed the data into a low-dimensional space for subsequent analysis. Traditional linear embedding techniques have solid theoretical foundations and are widely used, e.g. principal component analysis (PCA) [21, 44] and multidimensional scaling (MDS) [45, 5, 7] for datasets of a single modality, and canonical correlation analysis (CCA) [19, 1] for datasets of multiple modalities.

However, real datasets may exhibit nonlinear geometry, and discovering the underlying non-linearity can be beneficial for subsequent inference. Many manifold learning algorithms have been proposed to learn the intrinsic low-dimensional structure of nonlinear datasets, including Isomap [43, 38], locally linear embedding (LLE) [30, 29], Hessian LLE [9], Laplacian eigenmaps [2, 17], local tangent space alignment (LTSA) [51, 50], among many others. Most of them start with the assumption that the data are locally linear, explore the local geometry via the nearest-neighbor graph of the sample data, carry out transformation of the data based on the neighborhood graph, and eventually learn the low-dimensional manifold by optimizing some objective function. These nonlinear embedding algorithms usually serve as a preliminary feature extraction step that enables subsequent inference. They have been used successfully in object recognition and image processing.

In this paper, we consider the manifold matching task for datasets of multiple modalities, which is traditionally modeled by multiple dependent random variables. Classical methods for identifying the relationship among multiple random variables are still very popular in theory and practice, such as canonical correlation [19, 22, 16] and Procrustes transformation [35, 36, 13, 14]. However, it has become a much more challenging task to match real datasets of multiple modalities from disparate sources, such as the same document in different languages, an image and its descriptions, or networks of the same actors on different social websites.

There have been many recent endeavors regarding data fusion and manifold matching [24, 49, 48, 32, 28, 40, 34]. Similar to dimension reduction for datasets of a single modality, manifold matching can serve as a feature extraction step to explore datasets of multiple modalities, and has also been shown to help subsequent inference in object recognition [23], information retrieval [39], and transfer learning [27]. Furthermore,

**Corresponding author
e-mail: cshen6@jhu.edu (Cencheng Shen), jovo@jhu.edu (Joshua T. Vogelstein), cep@jhu.edu (Carey E. Priebe)

the matching task is important on its own and has been applied to explore multiple graphs and networks [26, 47, 25]. One such application is seeded graph matching, where two large networks are collected but only a percentage of training vertices have known correspondence. Thus, the remaining vertices need to be matched properly to uncover the potential correspondence.

Due to the success of nonlinear embedding algorithms for datasets of a single modality, it is often perceived that these algorithms can be combined into the matching framework to improve the matching performance when one or more modalities are nonlinear. A naïve procedure is to pick one nonlinear algorithm, apply it to each modality separately, and match the embedded modalities. But such a simplistic procedure does not always guarantee a good matching performance, since many nonlinear embedding algorithms only preserve the local geometry up to some affine transformation [12]. Furthermore, we will show in the numerical experiments that a direct matching of separate nonlinear embeddings can even deteriorate the matching performance when compared to linear embeddings.

To tackle the problem, we propose a manifold matching algorithm using shortest-path distance and joint neighborhood selection. By utilizing a robust distance measure that approximates the geodesic distance, and effectively combining the correspondence information into the embedding step, our proposed algorithm can significantly improve the matching quality from disparate data sources, compared to matching linear embeddings or separate nonlinear embeddings. All code and data are available on our website [1].

2. Manifold Matching

2.1. The Matching Framework

Suppose n objects are measured under two different sources. Then $X_l = \{x_{il}\} \in \Xi_l$ for $l = 1, 2$ are the actual datasets that are observed / collected, with $x_{i1} \sim x_{i2}$ for each i ($\sim$ means the two observations are matched in the context). Thus X_1 and X_2 are the two different views/modalities of the same underlying data. This setting is extendable to datasets of more than two modalities, but for ease of presentation we focus mainly on the matching task of two modalities.

Ξ_1 and Ξ_2 are potentially very different from each other, such as a flat manifold and its nonlinear transformation, an image and its description, or texts under different languages. A typical example is the social network, where many users have accounts on Youtube, Facebook, Twitter, etc. People often post different contents and connect with different groups on each website, such that data analysis of better quality is only possible when multiple accounts of the same person can be combined. Some accounts can be automatically matched, if the user already linked their accounts, or unique user information are filled and identified (like actual name, occupation), which provides a set of matched training data; but all the other accounts without full user information need to be matched by machine (as manual match is too expensive for millions of accounts), which provides a set of testing data from each website.

[1] https://github.com/cshen6/MMSJ

We assume $x_{il} \in \Xi_l$ is endowed with a distance measure Δ_l such that $\Delta_l(i, j) = dist(x_{il}, x_{jl})$. To match multiple modalities, we find two mappings $\rho_l : \Xi_l \to \mathbb{R}^d, l = 1, 2$ such that the mapped data $\hat{X}_l = \{\rho_l(x_{il})\}$ are matched into a common low-dimensional Euclidean space $\mathbb{R}^d$. A simple example of ρ_l can be MDS (e.g., classical MDS first doubly centers the distance matrices, followed by eigen-decomposition and keeping the top d eigenvalues and eigenvectors to yield the embedding) followed by CCA (find two orthogonal $d \times d$ transformation matrices for each data set to maximize their correlation), which is a linear embedding and matching procedure.

Once the mappings are learned, for any new observations $y_1 \in \Xi_1$ and $y_2 \in \Xi_2$ of unknown correspondence, the learned mappings ρ_l can be applied to match the testing observations in the low-dimensional Euclidean space, i.e., $\hat{y}_l = \rho_l(y_l) \in \mathbb{R}^d$. Ideally, a good matching procedure should be able to correctly identify the correspondence of the new observations, i.e., if the testing observations are truly matched in the context, the mapped points should be very close to each other in the common Euclidean space. If the testing observations are not matched, the mapped points should be far away from each other.

To evaluate a given matching algorithm, a natural criterion is the matching ratio used in seeded graph matching [26]. Suppose sufficient training observations are given to learn the mappings, and there exist some testing observations of unknown correspondence in each space. Assume that for each testing observation y_1 in Ξ_1, there is another testing observation $y_2 \in \Xi_2$ such that $y_1 \sim y_2$. Then they are correctly matched if and only if $\hat{y}_2$ is the nearest neighbor of $\hat{y}_1$ among all mapped testing data from Ξ_2. The matching ratio represents the percentage of correct matching of all testing observations, and thus a higher ratio indicates a better matching algorithm.

The matching ratio based on nearest neighbor is often conservative, and can be a very small number when matching disparate real datasets. In practice, it is often more interesting to consider all neighbors within a small threshold, or rank multiple neighbors up to a limit. To that end, the statistical testing power of the hypothesis $H_0 : y_1 \sim y_2$ considered in [28] is another suitable criterion, which takes the Euclidean distance $\|\hat{y}_1 - \hat{y}_2\|$ as the test statistic. To estimate the testing power for given data, we first split all observations into matched training data pairs, matched testing data pairs, and unmatched testing data pairs. After learning ρ_l from the matched training data and applying them to all testing data, the test statistic under the null hypothesis can be estimated from the matched testing pairs, and the test statistic under the alternative hypothesis can be estimated from the unmatched testing pairs. The testing power at any type 1 error level is directly estimated from the empirical distributions of the test statistic, and a higher testing power indicates a better manifold matching algorithm.

We used both the testing power and the matching ratio for evaluation in the numerical experiments. Note that if the critical value at a given type 1 error level is used as a distance threshold, the testing power equals the probability that the distance between the matched pair is no larger than the distance threshold. Since the matching ratio only considers the nearest

neighbor of the matched pair, the testing power is never smaller than the matching ratio.

2.2. Main Algorithm

The main algorithm is shown in algorithm 1, henceforth referred to as MMSJ.

Given the distance matrices Δ_l for $\{X_l, l = 1, 2\}$, we first construct an $n \times n$ binary graph G by k-nearest-neighbor using the sum of normalized distance matrices $\sum_{l=1}^{2} \frac{\Delta_l}{\|\Delta_l\|_F}$, i.e., $G(i, j) = 1$ if and only if $\sum_l \frac{\Delta_l(x_{il}, x_{jl})}{\|\Delta_l\|_F}$ is among the smallest k elements in the set $\{\sum_l \frac{\Delta_l(x_{il}, x_{ql})}{\|\Delta_l\|_F}, q = 1, \ldots, n\}$.

Next, for each modality X_l, we calculate the shortest-path distance matrix Δ_l^G based on the normalized Δ_l and the joint graph G, i.e., solve the shortest-path problem using the weighted graph $\frac{\Delta_l \circ G}{\|\Delta_l\|_F}$, where $\circ$ denotes the Hadamard product. Then we apply MDS to embed Δ_l^G into $\mathbb{R}^d$ for each l, followed by the Procrustes matching to yield the matched data $\hat{X}_l$, i.e., the Procrustes matching finds a $d \times d$ rotation matrix by

$$P = \arg \min_{P'P=I} \|P\tilde{X}_1 - \tilde{X}_2\|_F^2,$$

and sets $\hat{X}_1 = P\tilde{X}_1$ and $\hat{X}_2 = \tilde{X}_2$, where $\tilde{X}_l$ denotes the embedded data by MDS.

Once the manifolds are learned from the matched training data, algorithm 2 can match new testing data of unknown correspondence to the manifolds. Given the distance between testing and training $\Delta_l(y_l, X_l)$ and the shortest-path distances for the training data Δ_l^G, we first approximate the shortest-path distances $\Delta_l^G(y_l, X_l)$ by the respective nearest-neighbors of the testing data within each modality. Then the testing data y_l are embedded by MDS out-of-sampling (OOS) technique into $\mathbb{R}^d$ to yield $\tilde{y}_l$, followed by the Procrustes matching, i.e., $\hat{y}_1 = P\tilde{y}_1$ and $\hat{y}_2 = \tilde{y}_2$.

Note that algorithm 2 is applicable to testing data of arbitrary size, but for simplicity we present it for one testing observation from each modality.

To better visualize the process, we also summarize it in the flowchart of Figure 1.

2.3. Implementation Details

In this subsection, we discuss some implementation details of MMSJ, and the benchmarks we compare it with.

The algorithm starts with two distance matrices rather than the sample observations directly, which means MMSJ is directly applicable to multiple modalities with difference feature sizes, as long as a distance metric can be defined for each modality. Although there is no limitation on applying the MMSJ algorithm once the distances are given, the actual matching performance is clearly dependent on the choice of the metric. The most common choice is the Euclidean distance, or L^p metrics in general. Other similarity or dissimilarity measures may be more appropriate in certain domain, such as the cosine distance for text data (see Section 3.2), or suitable kernels for structured data [18].

The joint neighborhood graph ensures consistent neighborhood selection in the case of noisy or nonlinear modality, and

Algorithm 1 Manifold Matching using Shortest-Path Distance and Joint Neighborhood Selection (MMSJ)

Require: The distance matrices Δ_l for the matched datasets $\{X_l, l = 1, 2\}$, the neighborhood choice k, and the dimension choice d.

Ensure: The mapped datasets $\{\hat{X}_l \in \mathbb{R}^{d \times n}, l = 1, 2\}$, the shortest-path distance Δ_l^G, and the learned Procrustes transformation P.

function MMSJ(Δ_1, Δ_2, k, d)
  for $i, j := 1, \ldots, n$ **do** $G_{ij} \leftarrow \sum_l \frac{\Delta_l(x_{il}, x_{jl})}{\|\Delta_l\|_F}$ **end for**
  $G = \text{RANK}(G)$ ▷ rank distances within each row
  for $i, j := 1, \ldots, n$ **do** $G_{ij} \leftarrow \boldsymbol{I}(G_{ij} \le k)$ **end for**
  for $l := 1, 2$ **do**
    $\Delta_l^G = \text{SHORTESTPATH}(\frac{\Delta_l \circ G}{\|\Delta_l\|_F})$
    $\tilde{X}_l = \text{MDS}(\Delta_l^G, d)$ ▷ embedding into $\mathbb{R}^d$
  end for
  $[U, S, V] = \text{SVD}(\tilde{X}_2^T \tilde{X}_1)$
  $P \leftarrow UV^T$ ▷ Procrustes matching
  $\hat{X}_1 = P\tilde{X}_1$
  $\hat{X}_2 = \tilde{X}_2$
end function

Algorithm 2 Embed Testing Data based on MMSJ

Require: The distance vectors $\Delta_l(y_l, X_l)$, the shortest-path distance matrices Δ_l and the mapped data $\hat{X}_l$, the learned Procrustes transformation P, and the neighborhood choice k.

Ensure: The mapped testing observations $\hat{y}_l$.

function MMSJ2($\Delta_l(y_l, X_l), \Delta_l^G, \hat{X}_l, P, k$)
  for $l := 1, 2$ **do**
    $G_l = \text{RANK}(\Delta_l(y_l, X_l))$
    $\Delta_l^G(y_l, X_l) = \text{SHORTESTPATH}([\Delta_l^G | \Delta_l(y_l, X_l) \circ G_l])$
    $\tilde{y}_l = \text{MDS-OOS}(\hat{X}_l, \Delta_l^G(y_l, X_l))$
  end for
  $\hat{y}_1 = P\tilde{y}_1$
  $\hat{y}_2 = \tilde{y}_2$
end function

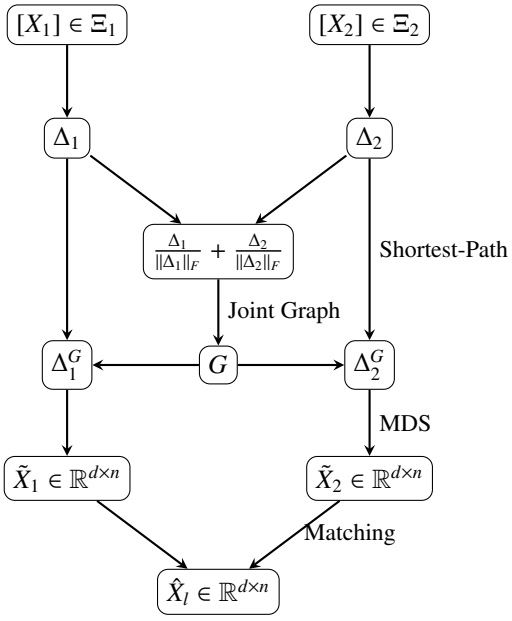

Fig. 1: Flowchart for Algorithm 1

is intuitively better than two separate neighborhood graphs for later matching when the correspondence is known. Alternatively, one may use a weighted sum of distances or a rank-based method to derive the joint neighborhood graph instead. Note that it is necessary for the distance matrices to be properly scaled so that the joint neighborhood selection is meaningful, and joint neighborhood should not be used for unknown correspondence like the testing data.

The shortest-path distance can recover the geodesic distance of isometric manifolds with high probability under certain sampling conditions [4, 38]. When used together with joint neighborhood, the shortest-path distance makes use of the correspondence information. Computationally, the shortest-path distance matrix can be effectively implemented by Floyd's algorithm or Dijkstra's algorithm [43], which can be further sped up by choosing a small set of landmark points [38, 3]. For embedding the testing data, we essentially treat the training data as landmark points and only compute the shortest path distances from the testing to the training.

Embedding the shortest-path distance matrices followed by matching is a standard procedure. Alternatively, one may match the embeddings by CCA or joint MDS, as discussed in [28, 11]. The advantages of MMSJ mostly lie in joint neighborhood and shortest-path distance; in fact, MMSJ always exhibits significant improvement, no matter which matching method to use. Thus we mainly consider the Procrustes matching for ease of presentation in the paper (CCA matching results are added to Appendix). Note that the testing data are embedded by out-of-sample MDS, which is a standard technique for MDS and kernel PCA [31, 3, 46], and more efficient than re-embedding all training and testing data. After the testing data are mapped onto the manifolds by the learned Procrustes matching, we may test the matched-ness of any two testing observations from different data sources as in section 2.1.

In terms of computation speed, suppose n is the total sample size, the running time complexity of MMSJ is $O(n^2)$, assuming the distance matrices are already given and the shortest-path distance step uses the fast landmark approximation. The only overhead is the distance matrix construction, which takes an additional $O(n^2 d)$, where d denotes the maximal feature dimension among all modalities.

To compare with MMSJ, we use the common procedure that embed each modality separately by MDS / Isomap / LLE / LTSA, followed by Procrustes matching. Note that MDS / Isomap / LLE can all operate directly on a distance matrix, but some nonlinear algorithms like LTSA have to start with the Euclidean data rather than a distance measure. Thus, if only the distance matrices are available, MDS is first used to embed the distance matrices into a Euclidean space $\mathbb{R}^{d'}$ with $d' \geq d$, followed by LTSA to embed into $\mathbb{R}^d$, then Procrustes matching.

3. Numerical Experiments

In this section, we demonstrate the numerical advantages of the proposed manifold matching algorithm, with MDS, Isomap, LLE, and LTSA as the benchmarks. Overall, we observed that our algorithm is significantly better than all the benchmarks in matching ratio and testing power in various simulated settings and real experiments.

3.1. Swiss Roll Simulation

The Swiss roll data from [43] is a 3D dataset representing a nonlinear manifold, but intrinsically generated by points on a 2D linear manifold. Figure 2 shows the 3D Swiss roll data with 5000 points in colors, along with its 2D embeddings by MDS, Isomap, and LLE. Clearly, MDS fails to recognize the nonlinear geometry while both Isomap and LLE succeed. However, the LLE embedding has a distorted geometry, while the Isomap embedding is similar to the underlying 2D linear manifold.

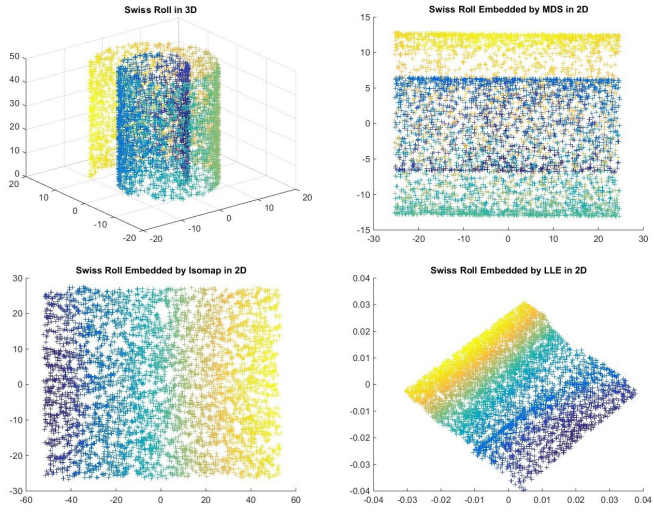

Fig. 2: The 3D Swiss roll dataset (top left), its 2D embedded data by MDS (top right), 2D embedding by Isomap at neighborhood size $k = 10$ (bottom left), and 2D embedding by LLE at $k = 10$ (bottom right).

For the first simulation, we matched the 3D Swiss roll with its underlying 2D linear manifold. A total of n points from the

3D Swiss roll were randomly generated to construct the first modality X_1, and the corresponding points on the underlying 2D linear manifold were taken as the second modality X_2. Thus X_1 and X_2 are matched training data with distinct geometries. Once the training data were matched, we embedded and applied the learned mappings to new testing observations y_1 and y_2 in each space.

We set the neighborhood size as $k = 10$, the dimension choice as $d = 2$, and generate another $n' = 100$ testing pairs to compute the matching ratio. We repeat the above process for 100 Monte-Carlo replicates, and show the average matching ratio in Figure 3(a) with respect to increasing size of the training data $n = 50, 100, \ldots, 1000$. The MMSJ algorithm exhibits a significant advantage over all other algorithms: it shows a better matching ratio from small sample size onwards, and achieves almost perfect matching as sample size grows.

Next, we checked the robustness of the manifold matching algorithms against noise, by adding white noise to the linear modality X_2. The noise was independently and identically distributed as $Normal(0, \epsilon I_{2\times 2})$, and the same testing procedure was applied to compute the matching ratio at each noise level. The results are plotted in Figure 3(b) with respect to the increasing noise level $\epsilon = 0, 1, 2, \ldots, 10$ at $n = 1000$. The MMSJ algorithm is clearly better than all the benchmarks as the noise level increases.

For the third simulation, we consider an outlier scenario, by randomly permuting a portion of the training data. For $\epsilon = 0, 0.01, \ldots, 0.1$, we randomly take ϵn training data from X_2 and permute their position, such that those training data are no longer matched with the corresponding points from X_1, i.e., there exists ϵn outliers in the training data. Fixing $n = 1000$, we apply the same testing procedure, and plot the matching ratio in Figure 3(c) with respect to the outlier percentage ϵ. The MMSJ algorithm is also robust against outliers, having better matching ratio throughout increasing ϵ; and all methods have insignificant matching ratio after the outlier percentage ϵ increases beyond 0.1, implying that the matching task may benefit significantly from excluding outliers prior to matching.

Note that the testing powers of all three simulations as well as the CCA matching results are shown in Appendix, which yields similar interpretations regarding the MMSJ superiority.

3.2. Wikipedia Articles Experiments

In this section, we applied the manifold matching algorithm to match disparate features of Wikipedia articles. The raw data contained 1382 pairs of articles from Wikipedia English and the corresponding French translations, within the 2-neighborhood of the English article "Algebraic Geometry". On Wikipedia, the same articles of different languages are almost never the exact translations of each other, because they are very likely written by different people and their contents may differ in many ways.

For the English articles and their French translations, a text feature and a graph feature were collected separately under each language. For the texts of each article, we used latent semantic indexing (LSI) (i.e., first construct a term-document matrix to describe the occurrences of terms in documents, then apply the low-rank approximation to the term matrix to 100 dimensions by singular value decomposition, see [8] for details) followed by cosine dissimilarity to construct two dissimilarity matrices TE and TF (representing the English texts and French texts). For the networks, two shortest-path distance matrices GE and GF (representing the English graph and French graph) were calculated based on the Internet hyperlinks of the articles under each language setting, with any path distance larger than 4 imputed to be 6 to avoid infinite distances and scaling issues.

Therefore, there existed four different modalities for pairs of Wikipedia articles on the same topic, making TE, TF, GE, and GF matched in the context. Furthermore, as the text matrices were derived by cosine similarity while the graph matrices were based on the shortest-path distance with imputation, the former probably had nonlinear geometries while the latter were linear from the view of our matching algorithm.

For each Monte-Carlo replicate, we randomly picked $n = 500$ pairs of training observations, 100 pairs of testing matched observations, and 100 pairs of testing unmatched observations for evaluation. The parameters were set as $k = 20$, $d = 10$, $d' = 50$ (for LTSA only), and the manifold matching algorithms were applied for every possible combination of matching two modalities. We performed a total of 100 Monte-Carlo replicates. The mean matching ratio is reported in Table 1, the estimated testing power is presented in Table 2 at type 1 error level 0.05, and the full power curves for some matching combinations are included in the Appendix.

Clearly, MMSJ achieves the best performance throughout all combinations. From the tables and figures, we further observe that without using shortest-path distance or joint neighborhood, separate nonlinear embeddings from LLE or LTSA are worse than the linear MDS embeddings in matching. Isomap does fairly well in the testing power, as it also uses shortest-path distance, but it can still be occasionally similar or slightly inferior to MDS in the matching ratio. Our proposed MMSJ algorithm is consistently the best manifold matching algorithm in both the testing power and the matching ratio throughout.

For the next experiment, we show that MMSJ algorithm is also robust against misspecification of parameters. The first two panels of Figure 4 show the MMSJ and Isomap testing powers (the best two algorithms in our matching experiments) for matching (TE, GE) against different choices of d and k, for which d ranges from 2 to 30 and k ranges from 10 to 30. It is clear that MMSJ is always better than Isomap in matching and attains similar testing power in a wide range of parameter choices; the best MMSJ testing power is 0.55 while the best Isomap testing power is 0.45. The same robustness holds for MMSJ under all other matching combinations.

Table 1: Wikipedia Documents Matching Ratio

Modalities	MMSJ	MDS	Isomap	LLE	LTSA
(TE, TF)	**0.2942**	0.2546	0.2003	0.1265	0.0491
(TE, GE)	**0.1209**	0.0675	0.0866	0.0143	0.0260
(TF, GF)	**0.0624**	0.0419	0.0522	0.0134	0.0144
(GE, GF)	**0.1347**	0.1280	0.1081	0.0157	0.0236
(TE, GF)	**0.0677**	0.0429	0.0560	0.0132	0.0138
(TF, GE)	**0.0946**	0.0545	0.0698	0.0132	0.0238

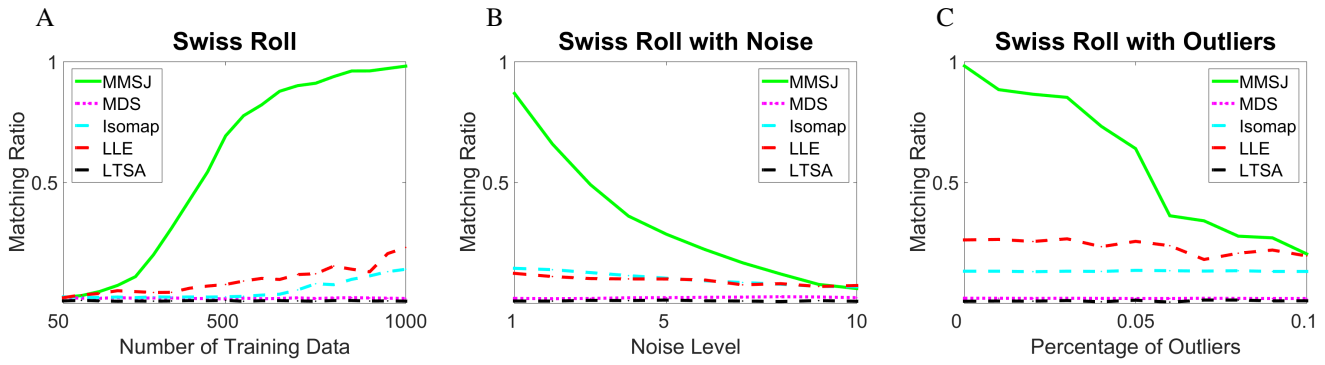

Fig. 3: Matching Ratio of 3D Swiss Roll versus its 2D Underlying Linear Manifold. (A) Matching Ratio with respect to Increasing Size of Training Data. (B) Matching Ratio with respect to Increasing Noise at $n = 1000$. (C) Matching Ratio with respect to Growing Number of Outliers at $n = 1000$.

Table 2: Wikipedia Documents Testing Power at Type 1 Error Level 0.05

Modalities	MMSJ	MDS	Isomap	LLE	LTSA
(TE, TF)	**0.8124**	0.4974	0.7476	0.3594	0.1930
(TE, GE)	**0.5184**	0.2563	0.4255	0.0948	0.1116
(TF, GF)	**0.2782**	0.1128	0.1877	0.0903	0.1028
(GE, GF)	**0.3108**	0.2141	0.2485	0.0961	0.1063
(TE, GF)	**0.3199**	0.1130	0.2141	0.0923	0.1021
(TF, GE)	**0.4464**	0.2114	0.3595	0.0943	0.1064

3.3. Brain Structural Networks

In this section, we assess the matching performance via brain structural networks. There are a total of $n = 109$ subjects, each with diffusion weighted magnetic resonance imaging (MRI) data. For the raw brain imaging data, we derived two different modalities. For each scan, (i) process diffusion and structural MRI data via MIGRAIN, a pipeline for estimating brain networks from diffusion data [15], (ii) compute the distance between brain networks using the semi-parametric graph test statistic [41, 42], then embed each graph into two dimensions and align the embeddings via a Procrustes analysis. The Euclidean distance is used on both modalities.

Therefore the first modality seems like a more faithful representation of the brain imaging, while the second modality is inherently a graph representation of the brain structure. Although these two modalities are merely different transformations of the same raw data, they are clearly distinct in many aspects such that there is no guarantee that one can recover their underlying geometry or succeed the matching task via machine learning algorithms.

For each Monte-Carlo replicate, we randomly picked n pairs of matched observations for training, with all remaining sample observations for testing. The parameters were set as $k = 7$, $d = 2$, $d' = 2$, with a total of 100 Monte-Carlo replicates. The mean matching ratio is shown in Figure 4(C) with respect to increasing size of training data. It is clear that MMSJ is the best matching method among all algorithms, and all matching ratios improve significantly as the training data size increases relative to the testing data size.

4. Concluding Remarks

In summary, we propose a nonlinear manifold matching algorithm using shortest-path distance and joint neighborhood selection. The algorithm is straightforward to implement, and achieves superior and robust performance. It is able to significantly improve the testing power and matching ratio when matching modalities of distinct geometries, and is robust against noise, outliers, and model selection. Our experiments indicate that the shortest-path distance and joint neighborhood selection are two key catalysts behind the improvement of the matching performance.

There are a number of additional potential extensions of this work. First, pursuing theoretical aspects of the manifold matching task is a very challenging but rewarding task: so far there is a very limited number of literatures even for manifold learning of single modality, and there is no guarantee that nonlinear transformation can always recover the linear manifold (the best known theoretical work is that shortest-path distance can recover the geodesic distance under certain class of nonlinear geometry for large sample size [37]). On the other hand, the task of matching multiple modalities is unique on its own. As a first step towards better theoretical understanding, we successfully proved in [33] that testing dependence via local correlations (which makes use of joint neighborhood information and local distance in a similar manner) can successfully detect almost all relationships as sample size grows large, which shall shed more lights into the consistency of the matching task and may further advance the MMSJ algorithm.

Second, MMSJ requires a pair of metrics (or distances) for each modality. In this work we assume such metrics are predefined by domain knowledge, or use the Euclidean distance otherwise. If an optimal metric can be reasonably selected for each modality, it is likely to further boost the performance of MMSJ. From another point of view, if we are given high-dimensional or structured data (say graphs or images), the MMSJ algorithm may further benefit from an appropriate feature selection down to certain landmark features [20, 6, 10]. This quest is a valuable future direction to work on.

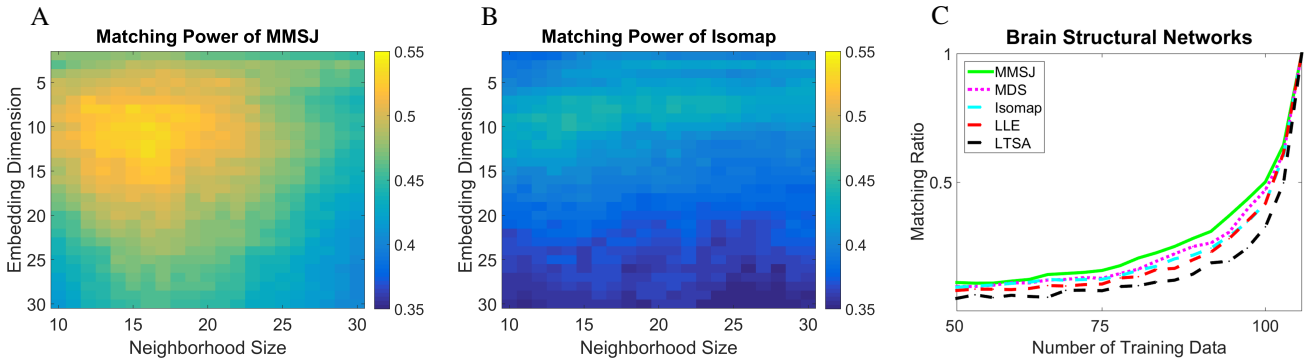

Fig. 4: (A) The Testing Power of MMSJ for Matching Wikipedia English Text and English Graph with respect to Different Dimension Choices and Neighborhood Sizes at Type 1 Error Level 0.05. (B) Same as (A) but for The Testing Power of Isomap. (C) Matching Ratio of Brain Structural Networks with respect to Increasing Size of Training Data.

Acknowledgment

This work is partially supported by the National Security Science and Engineering Faculty Fellowship (NSSEFF), the Johns Hopkins University Human Language Technology Center of Excellence (JHU HLT COE), and the XDATA program of the Defense Advanced Research Projects Agency (DARPA) administered through Air Force Research Laboratory contract FA8750-12-2-0303. This work is also supported by the Defense Advanced Research Projects Agency (DARPA) SIMPLEX program through SPAWAR contract N66001-15-C-4041 and DARPA GRAPHS N66001-14-1-4028.

The authors would like to thank the reviewers for their insightful and valuable suggestions in improving the exposition of the paper.

References

[1] Bach, F.R., Jordan, M.I., 2005. A Probabilistic Interpretation of Canonical Correlation Analysis. Technical Report. Department of Statistics, UC Berkeley.

[2] Belkin, M., Niyogi, P., 2003. Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation 15, 1373–1396.

[3] Bengio, Y., Paiement, J.F., Vincent, P., 2003. Out-of-sample extensions for LLE, Isomap, MDS, Eigenmaps, and Spectral Clustering, in: Advances in Neural Information Processing Systems, MIT Press. pp. 177–184.

[4] Bernstein, M., de Silva, V., Langford, J.C., Tenenbaum, J.B., 2000. Graph approximations to geodesics on embedded manifolds.

[5] Borg, I., Groenen, P., 2005. Modern Multidimensional Scaling: theory and applications. Springer-Verlag.

[6] Conte, D., Foggia, P., Sansone, C., Vento, M., 2004. Thirty years of graph matching in pattern recognition. International Journal of Pattern Recognition and Artificial Intelligence 18, 265–298.

[7] Cox, T., Cox, M., 2001. Multidimensional Scaling. Chapman and Hall.

[8] Deerwester, S., Dumais, S., Landauer, T., Furnas, G., Harshman, R., 1990. Indexing by latent semantic analysis. Journal of the American Society of Information Science 41, 391–407.

[9] Donoho, D., Grimes, C., 2003. Hessian eigenmaps: New locally linear embedding techniques for high-dimensional data, in: Proceedings of the National Academy of Arts and Sciences, pp. 5591–5596.

[10] Fiori, M., Sprechmann, P., Vogelstein, J., Mus, P., Sapiro, G., 2013. Robust multimodal graph matching: Sparse coding meets graph matching, in: Advances in Neural Information Processing Systems, pp. 127–135.

[11] Fishkind, D., Shen, C., Park, Y., Priebe, C.E., 2016. On the incommensurability phenomenon. Journal of Classification accepted.

[12] Goldberg, Y., Ritov, Y., 2008. Manifold learning: the price of normalization. Journal of Machine learning research 9, 1909–1939.

[13] Goldberg, Y., Ritov, Y., 2009. Local Procrustes for manifold embedding: a measure of embedding quality and embedding algorithms. Machine learning 77, 1–25.

[14] Gower, J.C., Dijksterhuis, G.B., 2004. Procrustes Problems. Oxford University Press.

[15] Gray Roncal, W., Koterba, Z.H., Mhembere, D., Kleissas, D.M., Vogelstein, J.T., Burns, R., Bowles, A.R., Donavos, D.K., Ryman, S., Jung, R.E., Wu, L., Calhoun, V.D., Vogelstein, R.J., 2013. MIGRAINE: MRI graph reliability analysis and inference for connectomics. Global Conference on Signal and Information Processing .

[16] Hardoon, D.R., Szedmak, S., Shawe-Taylor, J., 2004. Canonical correlation analysis: An overview with application to learning methods. Neural Computation 16, 2639–2664.

[17] He, X., Yan, S., Hu, Y., Niyogi, P., Zhang, H., 2005. Face recognition using Laplacianfaces. IEEE Transactions on Pattern Analysis and Machine Intelligence 27, 328–340.

[18] Hofmann, T., Scholkopf, B., Smola, A., 2008. Kernel methods in machine learning. The Annals of Statistics 36, 1171–1220.

[19] Hotelling, H., 1936. Relations between two sets of variates. Biometrika 28, 321–377.

[20] Jiang, J., Zheng, S., Toga, A., Tu, Z., 2008. Learning based coarse-to-fine image registration, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[21] Jolliffe, I.T., 2002. Principal Component Analysis. 2nd ed., Springer.

[22] Kettenring, J.R., 1971. Canonical analysis of several sets of variables. Biometrika 58, 433–451.

[23] Kim, T.K., Kittler, J., Cipolla, R., 2007. Discriminative learning and recognition of image set classes using canonical correlations. IEEE Transactions on Pattern Analysis and Machine Intelligence 29, 1005–1018.

[24] Lafon, S., Keller, Y., Coifman, R., 2006. Data fusion and multi-cue data matching by difusion maps. IEEE transactions on Pattern Analysis and Machine Intelligence 28, 1784–1797.

[25] Lyzinski, V., Fishkind, D., Fiori, M., Vogelstein, J.T., Priebe, C.E., Sapiro, G., 2016. Graph matching: Relax at your own risk. IEEE Transactions on Pattern Analysis and Machine Intelligence 38, 60–73.

[26] Lyzinski, V., Fishkind, D., Priebe, C.E., 2014. Seeded graph matching for correlated Erdos-Renyi graphs. Journal of Machine Learning Research 15, 3513–3540.

[27] Pan, S.J., Yang, Q., 2010. A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering 22, 1345–1359.

[28] Priebe, C.E., Marchette, D.J., Ma, Z., Adali, S., 2013. Manifold matching: Joint optimization of fidelity and commensurability. Brazilian Journal of Probability and Statistics 27, 377–400.

[29] Roweis, S.T., Saul, L.K., 2003. Think globally, fit locally: Unsupervised learning of low dimensional manifolds. Journal of Machine Learning Research 4, 119–155.

[30] Saul, L.K., Roweis, S.T., 2000. Nonlinear dimensionality reduction by locally linear embedding. Science 290, 2323–2326.

[31] Scholkopf, B., Smola, A., Muller, K., 1998. Nonlinear component analysis as a kernel eigenvalue problem. Neural Computation 10, 1299–1319.

[32] Sharma, A., Kumar, A., III, H.D., Jacobs, D., 2012. Generalized multiview analysis: A discriminative latent space, in: IEEE Conference on

Computer Vision and Pattern Recognition (CVPR).
[33] Shen, C., Priebe, C.E., Maggioni, M., Vogelstein, J.T., 2017. Discovering relationships across disparate data modalities. Submitted .
[34] Shen, C., Sun, M., Tang, M., Priebe, C.E., 2014. Generalized canonical correlation analysis for classification. Journal of Multivariate Analysis 130, 310–322.
[35] Sibson, R., 1978. Studies in the robustness of multidimensional scaling: Procrustes statistics. Journal of the Royal Statistical Society. Series B 40, 234–238.
[36] Sibson, R., 1979. Studies in the robustness of multidimensional scaling: Perturbation analysis of classical scaling. Journal of the Royal Statistical Society. Series B 41, 217–229.
[37] de Silva, V., Tenenbaum, J.B., 2002. Unsupervised learning of curved manifolds. Nonlinear Estimation and Classification .
[38] de Silva, V., Tenenbaum, J.B., 2003. Global versus local methods in nonlinear dimensionality reduction. Advances in Neural Informaiton Processing Systems 15, 721–728.
[39] Sun, M., Priebe, C.E., 2013. Efficiency investigation of manifold matching for text document classification. Pattern Recognition Letters 34, 1263–1269.
[40] Sun, M., Priebe, C.E., Tang, M., 2013. Generalized canonical correlation analysis for disparate data fusion. Pattern Recognition Letters 34, 194–200.
[41] Sussman, D.L., Tang, M., Fishkind, D.E., Priebe, C.E., 2013. A consistent dot product embedding for stochastic blockmodel graphs. Journal of the American Statistical Association 107, 1119–1128.
[42] Tang, M., Athreya, A., Sussman, D.L., Lyzinski, V., Park, Y., Priebe, C.E., 2016. A semiparametric two-sample hypothesis testing problem for random graphs. Journal of Computational & Graphical Statistics .
[43] Tenenbaum, J.B., de Silva, V., Langford, J.C., 2000. A global geometric framework for nonlinear dimension reduction. Science 290, 2319–2323.
[44] Tipping, M.E., Bishop, C.M., 1999. Probabilistic principal component analysis. Journal of the Royal Statistical Society, Series B 61, 611–622.
[45] Torgerson, W., 1952. Multidimensional Scaling: I. Theory and method. Psychometrika.
[46] Trosset, M.W., Priebe, C.E., 2008. The out-of-sample problem for classical multidimensional scaling. Computational Statistics and Data Analysis 52, 4635–4642.
[47] Vogelstein, J., Conroy, J., Lyzinski, V., Podrazik, L., Kratzer, S., Harley, E., Fishkind, D., Vogelstein, R., Priebe, C., 2015. Fast approximate quadratic programming for graph matching. PLOS ONE 10, e0121002.
[48] Wang, C., Liu, B., Vu, H., Mahadevan, S., 2012. Sparse manifold alignment, in: Technical Report, UMass Computer Science UM-2012-030.
[49] Wang, C., Mahadevan, S., 2008. Manifold alignment using Procrustes analysis, in: Proceedings of the 25th International Conference on Machine Learning.
[50] Zhang, Z., Wang, J., Zha, H., 2012. Adaptive manifold learning. IEEE Transactions on Pattern Analysis and Machine Intelligence 34, 253–265.
[51] Zhang, Z., Zha, H., 2004. Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. SIAM Journal on Scientific Computing 26, 313–338.

Appendix A. Supplementary Figures

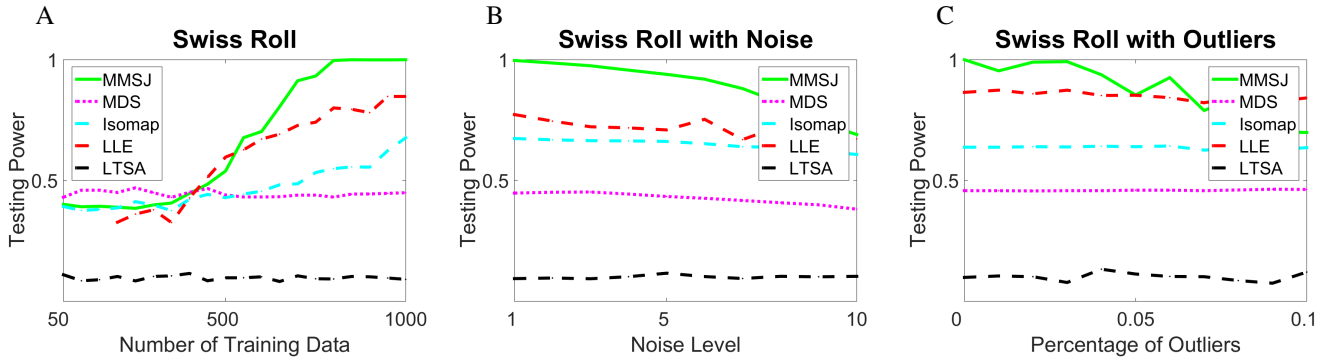

Fig. A1: Testing Power of 3D Swiss Roll versus its 2D Underlying Linear Manifold at type 1 error level 0.05, under the same setting as Figure 3. (A) Testing Power with respect to Increasing Size of Training Data. (B) Testing Power with respect to Increasing Noise at $n = 1000$. (C) Testing Power with respect to Growing Number of Outliers at $n = 1000$.

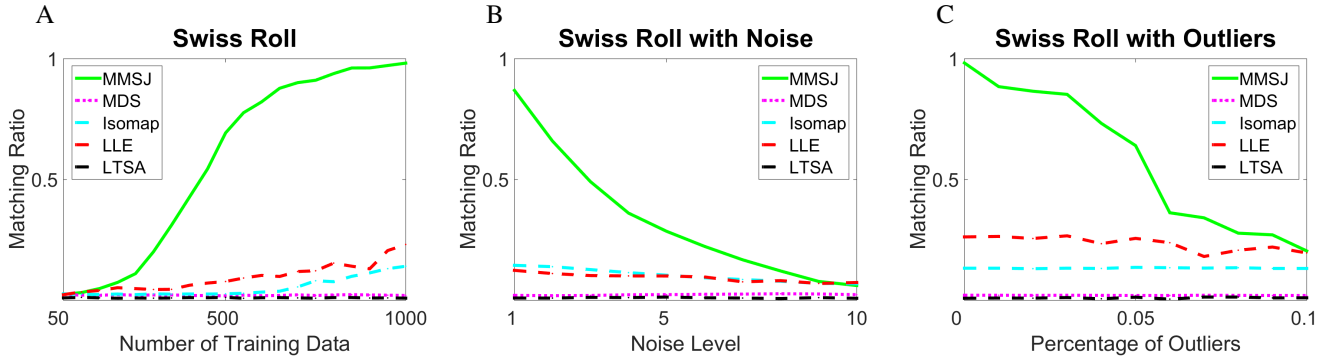

Fig. A2: Matching Ratio of 3D Swiss Roll versus its 2D Underlying Linear Manifold via CCA matching. (A) Matching Ratio with respect to Increasing Size of Training Data. (B) Matching Ratio with respect to Increasing Noise at $n = 1000$. (C) Matching Ratio with respect to Growing Number of Outliers at $n = 1000$.

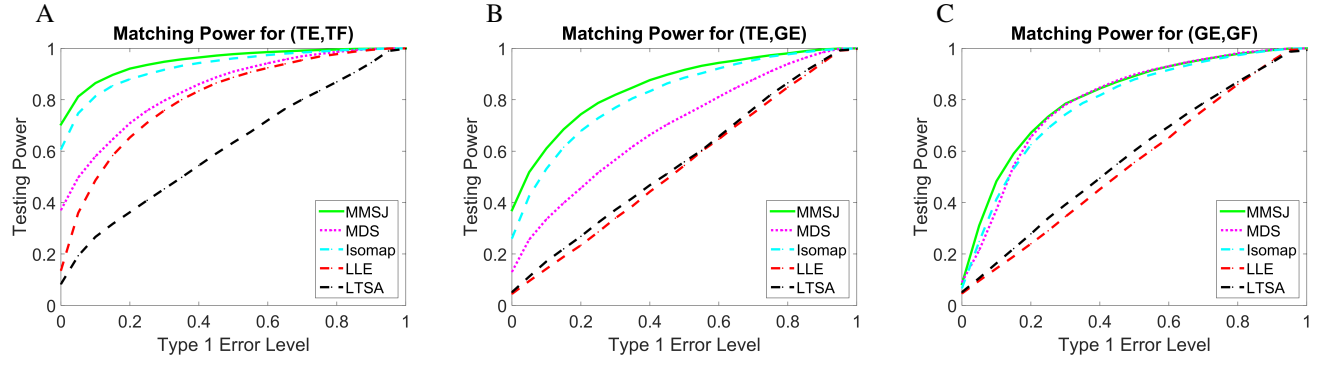

Fig. A3: Testing Powers of Wikipedia Datasets with respect to Increasing Type 1 Error Level. (A) Testing Power of Wikipedia English Text versus French Text. (B) Testing Power of Wikipedia English Text versus English Graph. (C) Testing Power of Wikipedia English Graph versus French Graph.