Consistency of dictionary-based Manifold Learning

Samson Koelle Uberduck
Hanyu Zhang Bytedance,Inc
Octavian-Vlad Murad University of Washington
Marina Meila University of Washington

Abstract

We analyze a paradigm for interpretable Manifold Learning for scientific data analysis, whereby one parametrizes a manifold with d smooth functions from a scientist-provided *dictionary* of meaningful, domain-related functions. When such a parametrization exists, we provide an algorithm for finding it based on sparse regression in the manifold tangent bundle, bypassing more standard, agnostic manifold learning algorithms. We prove conditions for the existence of such parameterizations in function space, and the first end-to-end recovery results from finite samples. The method is demonstrated on both synthetic problems and with data from a real scientific domain.

1 Introduction

Manifold Learning (ML) algorithms map high-dimensional data into a low-dimensional space by a learned function ϕ. Besides the computational saving, the manifold learned by ϕ can reveal the intrinsic variables that describe the behavior of the data source. This is why, in so many cases, scientists attempt to map manifold coordinates, as well as Principal Component (PC) representations, to variables known to be relevant to their domain. In Cavalli-Sforza et al. (1996), the PC of the genetic variation over Europe were matched to human migrations. Closer to our times, in Molecular Dynamic Simulations (MDS), data is produced that are medium and high dimensional, as shown in Figure 1a for the toluene molecule; 1b displays a single scientifically relevant function that models (approximately) the state space of the toluene molecule; it is an angle of rotation (*torsion*). This paper studies a semi-automatic method to do what scientists have been doing by visualization and manual inspection: find among a given set of functions $\mathcal{F}$ considered relevant by a scientist the small set that can parametrize a data manifold.

Koelle et al. (2022) proposed a method to combine the purely data driven approach of standard ML with "interpretations" of the estimated manifold based on prior knowledge of relevant scientific variables Specifically, the ManifoldLasso algorithm of Koelle et al. (2022) (which will be denoted MLasso here for brevity) will first map samples ξ_i, with $i = 1 : n$ from a manifold to an embeddig by $y_i = \phi(\xi_i) \in \mathbb{R}^m$; then each sample y_i receives new coordinates $f_{1:d}(\xi_i)$, where the functions $f_1, \dots f_d$ are selected from a predefined *finite* set of smooth functions $\mathcal{F}$, called a *dictionary*.

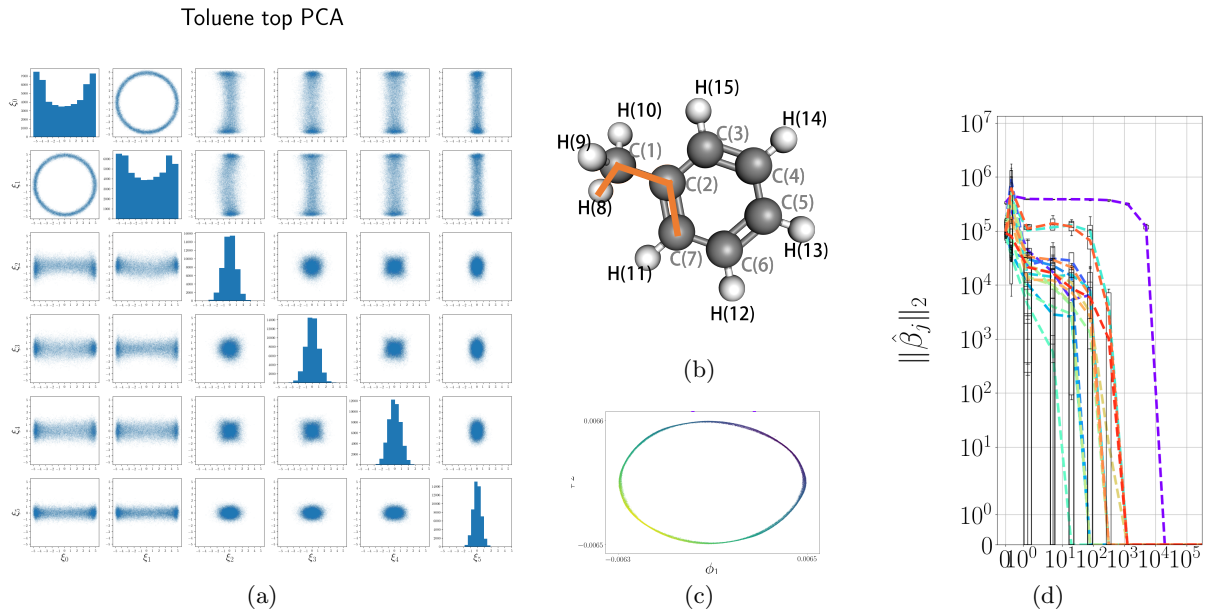
However, if the *coordinate functions* $f_{1:d}$ are unique in $\mathcal{F}$ for the embedding $\phi(\mathcal{M})$, then by a simple function composition one can see that the same $f_{1:d}$ will be unique coordinates for *any* other embedding of $\mathcal{M}$. In other words, the coordinates are a property of $\mathcal{M}$ itself, w.r.t. the dictionary $\mathcal{F}$.

Given $\mathcal{F}$, let f_S be the set $\{f_1, \dots f_d\}$. Here, we propose to (1) provide a direct algorithm, TSLasso, standing for *Tangent Space Lasso*, to recover f_S from the original high-dimensional samples $\xi_{1:n}$, and (2), to analyze the consistency of the recovery of f_S from $\mathcal{F}$. The output of TSLasso, f_S, will represent an embedding for $\mathcal{M}$.

Problem Statement Suppose data $\mathcal{D} = \{\xi_i, i \in [n]\}$ are sampled from a d-dimensional connected smooth[1] submanifold $\mathcal{M}$ embedded in the Euclidean space $\mathbb{R}^D$, where typically $D \gg d$. $\mathcal{M}$ has the Riemannian metric (Lee, 2003) induced from $\mathbb{R}^D$. We are also given a dictionary of functions $\mathcal{F} = \{f_j, j \in [p]\}$. All of the functions f_j are defined in the neighborhood of $\mathcal{M}$ in $\mathbb{R}^D$ and take values in some connected subset of $\mathbb{R}$. We require that they are smooth on $\mathcal{M}$ (as a subset of $\mathbb{R}^D$), and have analytically computable gradients in $\mathbb{R}^D$. Our goal is to select d functions in the dictionary, so that the mapping $f_S = (f_j)_{j \in S \subset \mathcal{F}}$ is a diffeomorphism on an open neighborhood $U \subset \mathcal{M}$ to $f_S(U) \subset \mathbb{R}^{|S|}$. If U covers $\mathcal{M}$ almost everywhere, f_S is then a *global* mapping with fixed number of functions. The learned

Proceedings of the 27th International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238. Copyright 2024 by the author(s).

[1] In this paper, by *smooth* manifold or function we mean of class C^l, $l \geq 1$, to be defined in Section 4.



(a) (b) (c) (d)

Figure 1: Finding a scientific interpretation for toluene MDS data. **a** pairwise scatterplots of first six coordinates in $\mathbb{R}^{50}$, after preprocessing (Supplementary Section 9). **b:** Toluene molecule. Scientists previously discovered that the torsion (relative rotation, marked in orange) of the peripheral (methyl) group governs the state space of the toluene molecule as a one dimensional manifold. **c:** Embedding of toluene data into $\mathbb{R}^2$ by Diffusion Maps, colored by the torsion labeled. **d:** Regularization path of TSLasso (with st. devs. over 25 replicates), showing that it selects the relevant torsion (purple path) from a dictionary of $p = 30$ possible torsions.

mapping f_S will be a *valid parametrization* of $\mathcal{M}$.[2]

The main technique, here as in Koelle et al. (2022), is to operate with gradient fields on $\mathcal{M}$. Section 2, introduces some background on gradient fields. In Section 3, we present algorithm TSLasso in detail. In Section 4, we provide sufficient conditions for selection consistency. Section 5 shows experimental results on synthetic and molecular dynamics data, while Section 6 discusses related work and interesting features of our approach.

2 Preliminaries: Gradients on Manifolds

In this section, we review gradient fields on manifolds, which play a central role in our algorithm. The reader is referred to Lee (2003) for the full treatment of this and related topics. Consider a $d-$dimensional manifold $\mathcal{M}$. At point ξ, its tangent space $\mathcal{T}_\xi\mathcal{M}$ can be viewed as the equivalence class of directions of infinitesimal curves passing ξ. For a smooth function $f : \mathcal{M} \mapsto \mathbb{R}$, its differential $Df : \mathcal{T}_\xi\mathcal{M} \mapsto \mathbb{R}$ is a linear map that generalizes directional derivatives in calculus in Euclidean space, characterizing how the value of f varies along different directions in $\mathcal{T}_\xi\mathcal{M}$.

When $\mathcal{M}$ is Riemannian with metric $\mathbf{g}$, the *gradient* of f is a collection of tangent vectors $\{\text{grad}\, f(\xi)$, for $\xi \in \mathcal{M}$, such that for all all $v \in \mathcal{T}_\xi\mathcal{M}$, $\langle \text{grad}\, f(\xi), v\rangle_{\mathbf{g}} = Df(v)|_\xi$. For example, under the usual Euclidean metric, a function $f : \mathbb{R}^D \mapsto \mathbb{R}$ has a gradient vector $\nabla f(\xi)$ at each point $\xi \in \mathbb{R}^D$ as defined in ordinary multivariate calculus.

For our problem, $\mathcal{M}$ is a $d-$dimensional manifold embedded in $\mathbb{R}^D$ with inherited metric. $\mathcal{T}_\xi\mathcal{M}$ can be identified as a $d-$dimensional linear subspace of $\mathcal{T}_\xi\mathbb{R}^D$, whose basis can be represented by an orthogonal $D \times d$ matrix $\mathbf{T}_\xi$. Let f be a smooth real-valued function, defined on a open neighborhood of $\mathcal{M}$. There are two points of view for f when it is restricted on $\mathcal{M}$: (i) as a function on (a subset of) $\mathbb{R}^D$ with gradient ∇f as usual, or, (ii), as a function on $\mathcal{M}$ with gradient field grad f given by the coordinate representation $\text{grad}\, f := \mathbf{T}_\xi^\top \nabla f$ (Lee, 2003). For some set S' of indices from $[p]$, grad $f_{S'}$ denotes the $d \times |S'|$ matrix with grad f_j, $j \in S'$, as columns.

[2]Note that, in differential geometry terminology, $(U \subseteq \mathcal{M}, f_S)$ is a coordinate *chart* for $\mathcal{M}$ and f_S^{-1} is called a *parameterization* of U. In this paper, we often refer to f_S as the 'parameterization', as f_S, f_S^{-1} are diffeomorphisms and are both representative. More details are in Supplement 7.

3 The TSLasso algorithm

The TSLasso algorithm is an embeddingless, direct version of the MLasso algorithm of Koelle et al. (2022). This section gives a self-contained description of TSLasso, while in Section 6 we discuss its relationship with MLasso.

The idea of the TSLasso algorithm is to replace the orthonormal bases $\mathbf{T}_\xi \in \mathbb{R}^{D\times d}$ of the manifold tangent spaces $\mathcal{T}_\xi\mathcal{M}$ with (possibly non-orthogonal) bases formed from gradients of d select dictionary functions. In this way, the original non-linear problem of selecting a functional approximation f_S to $\mathcal{M}$ from the dictionary $\mathcal{F}$, is transformed into a linear problem of selecting best local approximations in the tangent bundle.

If there is a subset f_S giving a valid parametrization in a neighborhood $U \subset \mathcal{M}$ of almost all points ξ, then f_S is a diffeomorphism. Hence, there is some mapping $g : f_S(U) \mapsto U$ such that $f_S \circ g$ is identity map on $f_S(U)$ and $g \circ f_S$ is the identity map on U. Thus, in coordinates, we can denote a matrix representation of grad $f_S(\xi)$ by $\mathbf{X}_{\xi,S} = \mathbf{T}_\xi^\top \nabla f_S(\xi) \in \mathbb{R}^{d\times d}$; further there is some matrix $\mathbf{B}_{\xi,S} \in \mathbb{R}^{d\times d}$ representing $\nabla g(f_S(\xi))$ such that

$$\mathbf{I}_d = \mathbf{X}_{\xi,S}\mathbf{B}_{\xi,S} \quad \text{for all } \xi \in U \tag{1}$$

according to the chain rule of differentiating function composition on manifolds.

For notation simplicity, we will write $\mathbf{X}_{iS}, \mathbf{B}_{iS}, \mathcal{T}_i\mathcal{M}$ as the corresponding quantities at point ξ_i when we are discussing a finite sample. Further, we denote by $\mathbf{X}_i \in \mathbb{R}^{d\times p} = \text{grad}\, f_{[p]}$, and $\mathbf{B}_i \in \mathbb{R}^{p\times d}$, a matrix so that $\mathbf{I}_d = \mathbf{X}_i\mathbf{B}_i$. Note that $\mathbf{B}_i$ can be obtained from $\mathbf{B}_{i,S}$ of (1) by completing it with zero columns. We also define $\mathbf{B}_{\cdot j} \in \mathbb{R}^{nd}$ as the vector formed by concatenating $\mathbf{B}_{i\{j\}}$, $i \in [n]$. Stacking $\mathbf{B}_{\cdot j}$ together forms $\mathbf{B} \in \mathbb{R}^{p\times nd}$.

3.1 Loss Function

We now seek a subset $S \subset [p]$ such that (1) only the corresponding nd vectors $\mathbf{B}_{\cdot j} : j \in S$ have non-zero entries and (2) each submatrix $\mathbf{X}_{iS}$ forms a rank d matrix. The observations immediately suggest minimizing the Frobenius norm of $\mathbf{I}_d - \mathbf{X}_i\mathbf{B}_i$ with joint sparsity constraints over columns of $\mathbf{B}_i$, induced over all data points.

$$J_\lambda(\mathbf{B}) = \frac{1}{2}\sum_{i=1}^{n} ||\mathbf{I}_d - \mathbf{X}_i\mathbf{B}_i||_F^2 + \frac{\lambda}{\sqrt{dn}}\sum_{j=1}^{p} ||\mathbf{B}_{\cdot j}||_2. \tag{2}$$

This optimization problem is a variant of Group Lasso (Yuan and Lin, 2006) that forces groups of coefficients of size dn to be zero simultaneously in the regularization path. It can be shown (Supplement 7.1) that this loss function is invariant to local tangent space basis.

3.2 Tangent Space Estimation

So far we have formulated our problem assuming we have access to the tangent space at each point $\xi \in \mathcal{M}$. However, this is rarely true. In practicee, the tangent spaces $\mathcal{T}_{\xi_i}\mathcal{M}$ at the data points must be estimated; we use a *Weighted Local Principal Component Analysis* (WL-PCA) method studied by Singer and Wu (2012); Chen et al. (2013); Aamari and Levrard (2018).

To perform WL-PCA, one must select a neighborhood radius r and identify $\mathcal{N}_i = \{i' \in [n], ||\xi_i - \xi_{i'}||_2 \leq r\}$ to be all neighbor points of ξ_i within Euclidean (in $\mathbb{R}^D$) distance r.

Each ξ_j in $\mathcal{N}_i$ is weighted by $K_{ij} = K(||\xi_i - \xi_j||/\epsilon)$, where $K()$ is a kernel function and ϵ is a tuning-parameter proportional to r, in the sense that kernel-values of pairs of non-neighboring points should be close to zero. Any C^2 positive monotonic decreasing function $K(u)$ with compact support is a valid kernel, examples including constant kernel $K(u) = 1_{[0,1]}(u)$, and Gaussian $K(u) = \exp(-u^2)1_{[0,1](x)}$ etc. We choose the Gaussian kernel in our experiments since it provides better tangent space estimation empirically, as it weights more on points that are close to where the tangent space is of interest.

Let $k_i = |\mathcal{N}_i|$ be the number of neighbors of point ξ_i and $\mathbf{\Xi}_i = [\xi_{i'}]_{i'\in\mathcal{N}_i} \in \mathbb{R}^{|\mathcal{N}_i|\times D}$ contain the neighbors of ξ_i. Denote a column vector of ones of length k by $\mathbf{1}_k$, and define the Singular Value Decomposition algorithm SVD$(\mathbf{X}, d)$ of matrix $\mathbf{X}$ as outputting $\mathbf{V}, \Lambda$, where Λ and $\mathbf{V}$ are the largest d eigenvalues and their corresponding eigenvectors. With these, tangent space estimation is performed as follows.

Algorithm 1 TangentSpaceBasis

1: **Input:** Local dataset $\mathbf{\Xi}_i$, intrinsic dimension d, kernel parameter ϵ
2: Compute local weights $K_{i,\mathcal{N}_i} = (K_{ij})_{j\in\mathcal{N}_i} \in \mathbb{R}^{k_i}$.
3: Compute weighted mean $\bar{\xi}_i = K_{i,\mathcal{N}_i}^\top \mathbf{\Xi}_i / (K_{i,\mathcal{N}_i}^\top \mathbf{1}_{k_i})$
4: Compute weighted local difference matrix $\mathbf{Z}_i = \text{diag}(K_{i,\mathcal{N}_i}^{\frac{1}{2}})(\mathbf{\Xi}_i - \mathbf{1}_{k_i}\bar{\xi}_i)$
5: Compute $\mathbf{T}_i, \Lambda \leftarrow \text{SVD}(\mathbf{Z}_i^\top \mathbf{Z}_i, d)$
6: **Output:** $\mathbf{T}_i$

3.3 The TSLasso Algorithm

We now present the full TSLasso algorithm. As explained at the beginning of Section 3, the original non-linear manifold parameterization recovery problem is turned into a collection of sparse linear problems in which the bases of individual tangent spaces are constructed from gradients of functions from dictionary $\mathcal{F}$. Tangent spaces at each point are estimated

in step 4, the gradients grad f_j of dictionary functions are obtained by projecting the $\mathbb{R}^D$ gradient $\nabla f_j(\xi_i)$ on to the estimated tangent space $\mathbf{T}_i$. Finally, with these gradients we form the objective function (2), which is minimized to obtain a sparse $\mathbf{B}$, and its support S.

Algorithm 2 TSLasso

1: **Input:** Dataset $\mathcal{D}$, dictionary $\mathcal{F}$, intrinsic dimension d, regularization parameter λ, radius parameter r, kernel parameter ϵ.
2: **for** $i = 1, 2, \dots n$ (or subset $I \subset [n]$) **do**
3: Compute $\mathcal{N}_i$ and $\mathbf{\Xi}_i$ using $\mathcal{D}, r$
4: Compute the orthonormal tangent space basis $\mathbf{T}_i \leftarrow$TangentSpaceBasis$(\mathbf{\Xi}_i, d, \epsilon)$
5: Compute $\nabla f_j(\xi_i)$ for $j \in [p]$.
6: Project onto tangent space $\mathbf{X}_i = \mathbf{T}_i^\top [\nabla f_j(\xi)]_{j\in[p]}$
7: **end for**
8: Solve for $\mathbf{B}$ by minimizing $J_\lambda(\mathbf{B})$ in (2).
9: **Output:** $S = \{j \in [p] : ||\mathbf{B}_{\cdot j}||_2 > 0\}$

3.4 Other considerations

Normalization The relative scaling of functions f_j will affect the solution of the Group Lasso problem (2), since functions with larger gradient norm will tend to have smaller $\|\mathbf{B}_{\cdot j}\|_2$. This can affect the support S recovered. Therefore, we compute $\gamma_j^2 = \frac{1}{n}\sum_{i=1}^n ||\nabla f_j(\xi_i)||_2^2$ and set $f_j \leftarrow f_j/\gamma_j$. This approximates normalization by $||\nabla f_j||_{L_2(\mathcal{M})}$. Since the normalization of f_j is done prior to projection, functions whose gradients are more parallel to the tangent space of $\mathcal{M}$ are favored. Gradient components perpendicular to $\mathcal{T}\mathcal{M}$, will be penalized by this normalization strategy.

Computation Note that we do not need to run TSLasso on the whole dataset in order to take advantage of all of our data, and can instead run on a subset $I \subset [n]$ such that $|I| = n'$. In particular, the search for the neighbors sets $\mathcal{N}_i$ is $O(Dnn')$, significantly less than the time to construct a full neighbor graph for an embedding. For each i, computing the local mean is $O(k_i D)$, and finding the tangent space is $O(k_i D^2 + k_i^3)$. Gradient computation runtime is $O(D)$. Projection is $O(dDp)$. For each Group Lasso iteration, the compute time is $O(n'mpd)$ (Meila et al., 2018).

Tuning We select the kernel bandwidth ϵ using the method of Joncas et al. (2017). For the regularization parameter λ, we apply binary search from $\lambda = 0$ to $\lambda_{\max} = \max_j(\sum_{i=1}^n (\| \operatorname{grad}_{T_i^\mathcal{M}} f_j(\xi_i))\|_2^2)^{1/2}$ (Koelle et al., 2022) and choose the λ value so that the cardinality of the selected support is d, the intrinsic dimension of $\mathcal{M}$. This is assumed to be given, or alternatively can be estimated as in Levina and Bickel (2004). The choice for n' should be of the order $d \log p$ (Candes and Tao, 2007).

In the next section, we introduce sufficient recovery conditions for the success of this approach.

4 Support Recovery Guarantee

In this section, we discuss the behavior of TSLasso theoretically. We will provide sufficient conditions so that TSLasso correctly selects certain group of functions in the dictionary with high probability w.r.t. sampling on the manifold and this probability converges to one if sample size tends to infinity.

Assumption 4.1. (manifold and dictionary)

1. $\mathcal{M}$ is a d-dimensional $C^\ell, \ell \geq 1$ compact manifold with reach $\tau > 0$ embedded in $\mathbb{R}^D$.
2. Data $\{\xi_i\}_{i=1}^n$ are sampled from some probability measure P on $\mathcal{M}$ that has a Radon-Nikodym derivative $\pi(\xi)$ with respect to the Hausdorff measure. There exist two positive constants $\pi_{\min}, \pi_{\max}$ such that $0 < \pi_{\min} \leq \pi(\xi) \leq \pi_{\max}$ for all $\xi \in \mathcal{M}$.
3. Dictionary $\mathcal{F} = \{f_j(\xi) : j \in [p]\}$ contains C^1 functions defined on a neighborhood of $\mathcal{M}$ in $\mathbb{R}^D$. Further assume that $\delta := \inf_{\xi\in\mathcal{M}} \min_{j\in S} ||\nabla f_j(\xi)||_2 > 0$ and denote $\Gamma := \sup_{\xi\in\mathcal{M}} \max_{j\in[p]} ||\nabla f_j(\xi_i)||_2$.
4. $S \subset [p], |S| = d$ is the only subset such that $\operatorname{rank} f_S = d$ a.e. on $\mathcal{M}$ w.r.t. Hausdorff measure.

Assumption 1 on manifold and 2 on sampling are common in the manifold estimation literature (e.g. Aamari and Levrard (2018)). Assumption 3 restricts the smoothness of all dictionary functions and ensures that the functions in S do not have critical points on $\mathcal{M}$ as a function *on* $\mathbb{R}^D$. Also, notice that $\Gamma < \infty$ by the compactness assumption of $\mathcal{M}$.

Besides, we also assume TSLasso is performed as follows:

Assumption 4.2. Suppose tangent spaces are estimated by WL-PCA in Section 3.2 using neighborhood radius choice r_n and Gaussian kernel with bandwidth $\epsilon_n \propto r_n$, and normalization on dictionary is performed as in Section 3.4.

Now we are ready to prove support recovery consistency under suitable conditions. Let $\hat{\mathbf{B}}$ be the solution of problem (2) and $S(\hat{\mathbf{B}})$ be the nonzero rows of $\hat{\mathbf{B}}$. We will show that the probability of $S(\hat{\mathbf{B}}) = S$ converges to 1 as n increases. We start by defining the matrix $\tilde{\mathbf{X}}_\xi$ whose j-th column is $\mathbf{X}_{\xi,\cdot j}/||\nabla f_j(\xi)||_2$. Correspondingly we can define $\tilde{\mathbf{X}}_{\xi,S}$ as the submatrix of $\tilde{\mathbf{X}}_\xi$ with columns in S. Let $\mathbf{G}_{\xi,S} = \operatorname{diag}\{||\nabla f_j(\xi)||_2\}_{j\in S}$ and

define

$$\mu_S = \sup_{\xi \in \mathcal{M}, j \in S, j' \notin S} |\tilde{\mathbf{X}}_{\xi,\cdot j}^\top \tilde{\mathbf{X}}_{\xi,\cdot j'}| \ , \tag{3}$$

$$\nu_S = \sup_{\xi \in \mathcal{M}} ||(\tilde{\mathbf{X}}_{\xi,S}^\top \tilde{\mathbf{X}}_{\xi,S})^{-1} - \mathbf{G}_{\xi,S}^2||_2 \ . \tag{4}$$

The parameter μ_S can be thought of as a renormalized incoherence between the functions in S and those not in S; ν_S is a internal colinearity parameter, which is small when the columns of $\mathbf{X}_S(\xi)$ are closer to orthogonality and the gradient of functions in S are more parallel to the tangent space. We also define

$$\phi_S = \sup_{\xi \in \mathcal{M}} \max_{j \in S} ||\nabla f_j(\xi)||_2 \ , \tag{5}$$

$$\delta_S = \inf_{\xi \in \mathcal{M}} \min_{j \in S} ||\nabla f_j(\xi)||_2 \ , \tag{6}$$

which provide upper and lower bounds of the Euclidean gradient of functions in S. The following proposition provides a sufficient condition on $\mu_S, \nu_S, \phi_S, \delta_S$ such that S can be recovered consistently by TSLasso .

Proposition 4.3. *Suppose Assumptions 4.1 and 4.2 hold. If* $(1 + \nu_S/\delta_S^2)\mu_S\phi_S\Gamma d < 1$ *then there are constants* C, N_0 *depending only on* $\mathcal{M}, \pi_{\min}, \pi_{\max}$ *such that when* $n > N_0$ *and neighborhood radius selection* $r_n = C(\log n/(n-1))^{\frac{1}{d}}$*, it holds that*

$$Pr(S(\widehat{\mathbf{B}}) \subset S) \geq 1 - 4(\frac{1}{n})^{\frac{2}{d}} \ . \tag{7}$$

If it further holds that $\sqrt{d}\tilde{\nu}_S < \delta_S^2$ *and* $\lambda(1 + \tilde{\nu}_S/\delta_S^2) < \frac{1}{2}\sqrt{n - \frac{\sqrt{d}n\tilde{\nu}_S}{\delta_S^2}}$*, the same probability bound in 7 holds for the event* $S(\widehat{\mathbf{B}}) = S$

The proof is contained in the supplementary material. The main idea is first to find a sufficient condition so that given correct gradient of each function TSLasso can find the correct support, assuming correct estimation of the tangent space. Then we consider this condition in the case where $\mathcal{T}_{\xi_i}\mathcal{M}$ is estimated from data and obtain the guarantee by the fact that tangent spaces can be consistently estimated.

There are several remarks we would like to make on this result.

First, this result shows that the rate only depends on the intrinsic dimension d of the underlying manifold, and is *independent of* the ambient dimension D.

Second, to achieve the exact recovery, λ can must scale as $\sqrt{n}$. For a constant λ value, as sample size increases, the penalty term will decrease and the regularization effect will be reduced.

Third, the noise structure for this problem is not the same as a general Group Lasso problem since the source of noise is estimation of tangent space. Therefore the noise is neither isotropic nor Gaussian. In fact, since we are sampling *on the manifold*, there is no noise level parameter that appears as in standard Lasso literature. In a simulation experiment, we explore the behavior of our method on noisy settings and our method is robust against the case when data are sampled with certain level of noise. The proof technique used here can be generalized very easily to data under additive noise or cluttered noise, as defined in Aamari and Levrard (2018).

Fourth, there are some differences to be noted of this recovery result compared with classical recovery guarantees in Group Lasso type problems in e.g. Wainwright (2009), Obozinski et al. (2011), Elyaderani et al. (2017). We cannot adopt directly the usual assumption in Lasso literature that each column of $\mathbf{X}$ has unit norm, considering the normalization in Section 3.2. Also, the asymptotic regime we are considering here is only $n \to \infty$. Although we are using a Group Lasso type optimization problem, the dimension p is fixed since we only consider the fixed dictionary. There is no other conditions between p and n in our result, as required in many literature.

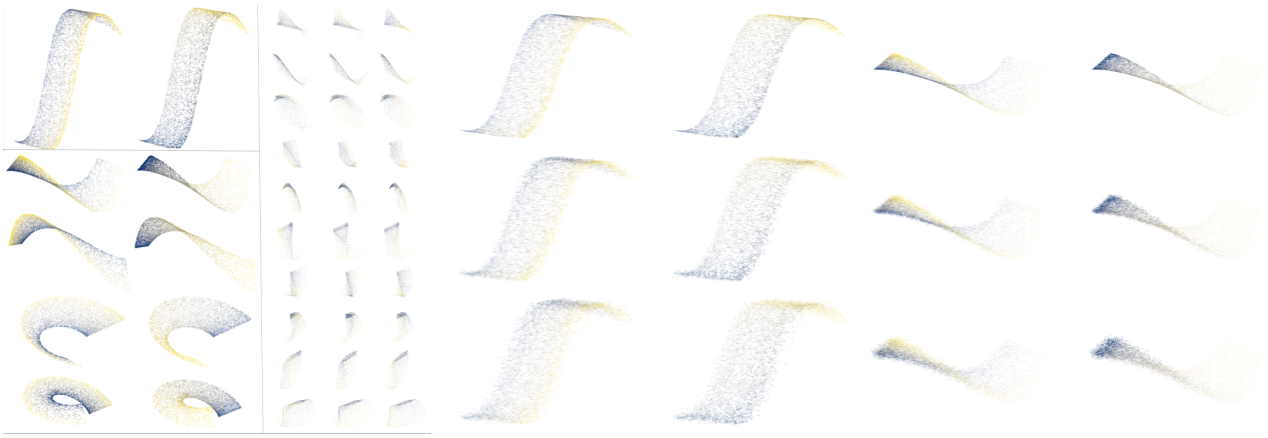
Finally, the proof technique used here could be used to develop the sample consistency of the MLasso algorithm in (Meila et al., 2018), which seeks explanations with physical meanings of a given embedding functions from dictionary functions, as long as the embedding functions to be explained is well-conditioned and the Riemannian metric is consistently estimated.

5 Experiments

We illustrate the behavior of TSLasso on synthetic and real data. For all of the experiments, the data consist of n points in D dimensions. TSLasso is applied to a uniformly random subset of size n' using p dictionary functions, and this process is repeated ω number of times. Note that the entire data set is used for tangent space estimation. The local tangent space kernel bandwidth ϵ_N is estimated using the algorithm of Joncas et al. (2017). Parameters are summarized in Tables 1 and 2. The experiments were performed on a 16 core Linux Debian Cluster with 768 gigabytes of RAM. Code and data are available at `github.com/sjkoelle/montlake`.

5.1 Experiments on Synthetic Data

The purpose of these experiments is to demonstrate the performance of TSLasso and explore its empirical limits in instances of controlled difficulty, to examine its robustness to ambient space off-manifold noise, and



(a) (b) (c)

Figure 2: **a**: Low dimensional representations of the functions $\mathcal{M}_1$(top left), $\mathcal{M}_2$(bottom left), $\mathcal{M}_3$(right). Each column j colors the points as a function of the manifold's j-th coordinate. **b, c**: Some of the same representations at different noise levels. The rows correspond to noise levels $\sigma = 0.0, 0.025, 0.05$. The second row($\sigma = 0.025$) is the highest level of noise for which the recovery of TSLasso is near-perfect, while the last row($\sigma = 0.05$) represents the lowest noise level at which all algorithms tested fail.

to compare these with the performance of MLasso. In particular, since the two algorithms both perform perfectly in our real data experiments(see Section 5.2), our synthetic datasets consist of 10 more difficult and noisy manifolds where we can display that TSLasso is indeed more robust than MLasso.

To generate our synthetic datasets, we developed a standalone package `symanifold`[3], which allows the user to symbolically and efficiently define, invert, differentiate, compose, and evaluate non-linear functions and their gradients. We use `symanifold` to create complex manifolds by symbolically defining non-linear injective functions which embed neighborhoods of $\mathbb{R}^d$ into $\mathbb{R}^D$. By inverting these functions, we obtain the true coordinate functions $f_{1:d}$ which are *non-linear, multivariate* combinations of the manifold coordinates. Additionally, we create "fake" (non-coordinate) functions $f_{d+1:p}$ which interact non-linearly with one of the true functions and have varying gradients over the manifolds, thus making recovery non-trivial. Using `symanifold`, we also symbolically compute the gradients ∇f_i of these functions. Finally, we sample data points on the manifold(with or without Gaussian noise with σ standard deviation), evaluate their gradients, and estimate theoretical condition numbers and difficulty measures such as μ_S, ν_S. We give a comprehensive description of the synthetic datasets in Supplement Section 8.

Experiment Setups The synthetic data consist of three manifolds $\mathcal{M}_{1,2,3} \subseteq \mathbb{R}^D$, with $D = 48$, intrinsic dimensions $d = 2, 2$ and 3 respectively, and dictionaries $\mathcal{F}_{1,2,3}$ with $p = d + 36$ functions. The difficulty of the recovery problem measured by μ_S and ν_S is recorded in Table 1. These values place us outside the theoretical recovery conditions. Finally, noise was added to $\xi_{1:n}$ until recovery failed.

Dataset	n	d	D	p	n'	ω	μ_S	ν_S
$\mathcal{M}_1$	5000	2	48	38	500	25	0.99	33
$\mathcal{M}_2$	5000	2	48	38	500	25	0.55	46
$\mathcal{M}_3$	5000	3	48	39	500	25	0.97	51

Table 1: Parameters for the synthetic data experiments

Dataset	n	d	D	p	n'	ω	ϵ_N	N_a
Eth	50000	2	50	12	100	25	3.5	9
Mal	50000	2	50	12	100	25	3.5	9
Tol	50000	1	50	30	100	25	1.9	15

Table 2: Parameters for the real data experiments: Eth(Ethanol), Mal(Malonaldehyde) and Tol(Toluene)

Results on synthetic data TSLasso was compared with MLasso using either UMAP(McInnes et al. (2018a) or Diffusion Maps(Coifman and Lafon (2006)) as the embedding algorithm. Figure 4 shows the recovery success for the three algorithms at various noise levels. We see that, despite the difficulty of the problem, we see that TSLasso behaves robustly. While MLasso_UMAP is mostly successful at low and moderate noise (see also Section 8 in Supplement), TSLasso

[3] `symanifold` can also be found at `github.com/sjkoelle/montlake`

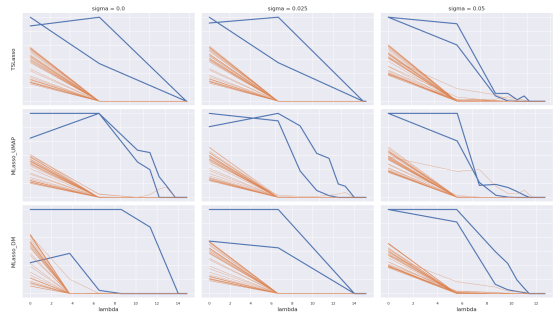


Figure 3: The regularization paths obtained by the three algorithms tested on $\mathcal{M}_2$ and averaged over 25 trials. The x-axis represents the value of λ, the strength of the Group Lasso regularization. An orange line("fake" function) crossing on top of a blue line(a true coordinate function) indicates a consistent failure to fully recover the support. Note that for $\sigma \leq 0.025$, the binary search for λ ends after just 1 step for TSLasso and MLasso_DM, and after about 6 steps for MLasso_UMAP, in failure. Complete regularization paths are in Supplement Section 8

is all around more robust, particularly on the more difficult $\mathcal{M}_3$ manifold. [4] As expected, TSLasso is about 20-50% faster than MLasso in the stable recovery cases(see Supplement Section 8). Furthermore, we expect the speedup of TSLasso relative to MLasso to increase as n increases.

5.2 Experiments on scientific MDS data

MDS simulations dynamically generate atomic configurations which, due to interatomic interactions, exhibit non-linear, multiscale, non-i.i.d. noise, as well as nontrivial topology and geometry. That is, they lie near a low-dimensional manifold Das et al. (2006). Additionally, the dictionaries (consisting of torsions and angles) are often known a-priori, but not the functions therein that parametrize the data manifold. Thus they are an appropriately challenging testbed for TSLasso.

From the scientific point of view, MDS are a heavily used tool, with Mega-hours of HPC devoted to them. Automatically finding scientifically meaningful parametrizations provides scientific insight about the simulated systems. The parametrizations $f_{1:d}$ can be used to further accelerate sampling. Parametrization such as the ones sought by TSLasso allow scientists to communicate in high level language about different experiments. For example, many MDS for small molecules will result on a d-torus; then, if two experiments in different conditions both result in tori, how do we know if they are different tori, or "the same"?

Experiment Setup In Supplement Section 9.1 we describe the preprocessing performed on the MDS data. This produces point clouds in $D = 50$ dimensions, and dictionaries consisting of torsions (rotation angles), such as in Figure 1b. We include in the dictionaries all torsions implied by the *bond diagram*, i.e. all the relative rotations of molecule parts connected by a segment in Figurs 1b, 5a 5c.[5] This choice reflects a priori information about molecular structure garnered from historical work. Building a dictionary based on this structure is akin to many other methods in the field (Krenn et al., 2020; Xie et al., 2019). Since original angular data featurization is an overparameterization of the shape space, one cannot use automatically obtained gradients in TSLasso. We therefore project the gradients prior to normalization on the tangent bundle of the shape space as it is embedded in $\mathbb{R}^D$.

Results on MDS Data For these MDS data, the ground truth is known, and in each case, the torsions that parametrize the MDS manifolds are present in the dictionaries. The TSLasso was run on a sample of $n = 50,000$ MDS data, from which a random subsample of size n' was used for the optimization problem 2. The subsampling was repeated $\omega = 25$ times with replacement. Figures 1d, 5b and 5d show the results of the TSLasso algorithm on these data. For toluene and ethanol, recovery of the relevant torsions is successful in all runs, while for malonaldehyde is succeeds in 24 out 25 runs.

These results can be compared with the experimental results published MLasso_DM on same size data sets, same dictionaries and similar setups in Koelle et al. (2022) (Section 7.4.1). We see that the TSLasso and MLasso_DM have qualitatively and quantitatively almost identical behavior (e.g. similar shapes for the regularization paths, MLasso_DM success rate is 25/25 on all data sets, etc.). This demonstrates that, in practice as well as in theory, TSLasso can substitue MLasso in recovering the set S of coordinates from a dictionary.

In the Supplement Section 9 the recovered parametrizations f_1, f_2 for ethanol and malonaldehyde are superimposed on a Diffusion Maps embedding. It is visible that, for malonaldehyde, recovering the correct torsions by visual inspection is far from trivial, thus the intervention of a quantitative algorithm can make a scientist's task easier.

[4] We believe that the poor performance of MLasso_DM stems from the manifolds' aspect ratio, which causes the Diffusion Maps algorithm to create rank deficient mappings (Chen and Meilă, 2019).

[5] More precisely, this dictionary consist of all equivalence classes of 4-tuples of atoms implicitly defined along the molecule skeletons.

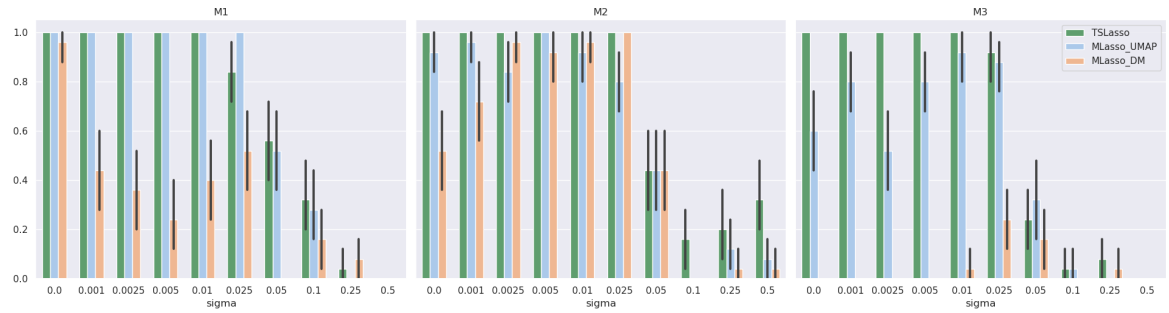


Figure 4: Successful support recoveries as proportion of 25 independent trials for TSLasso, MLasso_DM, and MLasso_UMAP. The error bars represent 90% confidence intervals. Note that MLasso_DM fails completely on $\mathcal{M}_3$ and has overall inconsistent results, while MLasso_UMAP's performance drops on the $\mathcal{M}_3$ data. This contrasts with TSLasso which achieves consistent results on all manifolds for noise levels up to $\sigma = 0.025$

We also point out that, in our experiments, the subsampled size $n' = 100$ is only around 0.2% of the whole dataset and in almost all replicates this subsample is sufficient to obtain a valid parametrization. Tangent space estimation is only needed for these points. Therefore bypassing the usual manifold embedding procedure (on the whole dataset) we are able to obtain interpretable embeddings with fewer samples and in a shorter time.

6 Discussion and Related Work

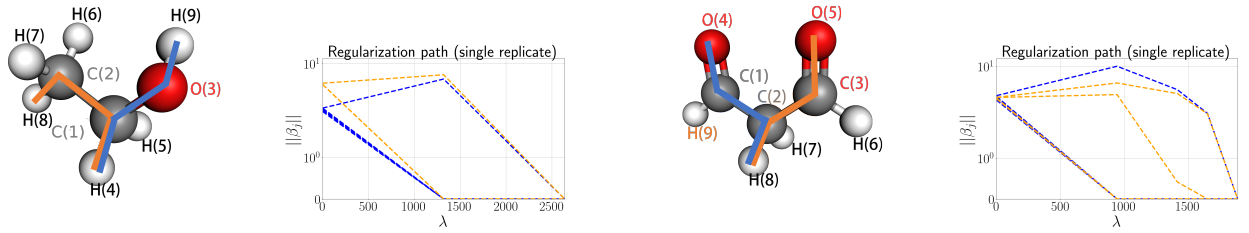
This paper is both about learning manifolds and about finding interpretable low dimensional descriptors for a scientific domain. The latter problem is very old, however manifold learning, too, has by now a rich history. It is worth pausing to see how these two problems differ, and how they are similar. Generic manifold learning is agnostic – the goal is to estimate a manifold from data, with only standard assumptions about the data generating process. The former problem, however, is generally not. The descriptors must be expressible in the language of the domain to be useful. Thus, the TSLasso/MLasso algorithms operate in this knowledge-rich framework. On the other hand, once the data is well approximated by a fixed set of smooth descriptors, we have *de facto* learned a manifold (locally or globally).

In this paper, by TSLasso we have presented a simple algorithm that is both a data interpretation and a manifold learning algorithm. This algorithm is backed by a end-to-end support recovery guarantees under standard statistical assumptions. To our knowledge, this is the first proof of Lasso consistency in function space, on a manifold. The proof developed extends naturally to other functional regression problems on manifolds. We expect that this proof can be extended to MLasso too, depending on regularity conditions on the embedding ϕ.

From the scientific point of view, parametrizing the data with a dictionary can be used to summarize scientific data in the language of the domain and to compare embeddings from different sources. From the point of view of manifold learning, a functional form f is smooth, invertible, and can be used to derive out-of-sample extensions and to interrogate mechanistic properties of the analyzed system. Finally, from the point of view of dictionary size, our method is flexible with respect to a range of non-linearities.

These features of TSLasso are shared with MLasso, but contrast with standard approaches in non-linear dimension reduction. Parametrizing high-dimensional data by a small subset of smooth functions has been studied outside the context of *autoencoders* (Goodfellow et al., 2016). Early work on parametric manifold learning includes Saul and Roweis (2003) and Teh and Roweis (2002), who proposed a mixture of local linear models whose coordinates are aligned. In a *non-parametric* setting, LTSA (Zhang and Zha, 2004) also gives a global parametrization by aligning locally estimated tangent spaces. However, both the parametric and non-parametric methods above produce embeddings ϕ that are abstract in the sense that they do not have a concise, interpretable functional form. In this sense, we draw a parallel between our approach and *factor models* (Yalcin and Amemiya, 2001).

Group Lasso type regression for gradient-based variable selection was previously explored in Haufe et al. (2009) and Ye and Xie (2012), but both have a simpler group structure, and are not utilized in the setting of dimension reduction. More recently, so-called *symbolic regression* methods such as Brunton et al. (2016), Rudy et al. (2019), and Champion et al. (2019) have been used for linear, non-linear, and machine-learned systems, respectively, and these methods may regarded as univariate relatives of our approach, since they are concerned with dynamics through time, while we consider



(a) (b) (c) (d)

Figure 5: Results for molecular dynamics data. 5a, 5c show bond diagrams for ethanol and malonaldehyde, respectively; 5b, 5d are regularization paths for one run of TSLasso on ethanol and malonaldehyde data, respectively. The colors correspond to the torsions in 5a and 5c.

the data manifold independently of time.

We also draw several distinctions between the TSLasso method and the MLasso method in Meila et al. (2018). First MLasso uses the same essential idea of sparse linear regression in gradient space, but in order to explain individual embedding coordinate functions, while in TSLasso there is no consistent matching between unit vectors in $\boldsymbol{I_d}$, and so can only provide an overall regularization path, rather than one corresponding to individual tangent basis vectors. Second, TSLasso method dispenses *with the entire Embedding algorithm*, Riemannian metric estimation, and pulling back the embedding gradients steps in MLasso , while providing almost everything a user can get from MLasso . Apart from simplification, TSLasso can also be more robust, since MLasso's success is predicated on the "success" of the embedding.

Acknowledgements

This work was partially supported by NSF DMS 2015272, NSF DMS 1810975, NSF DMS PD 08-1269, NSF IGERT 1258485, the Moore-Sloan Foundation, the UW eScience Institute, and a Simons Fellowship to MM from the Institute for Pure and Applied Mathematics (IPAM). Part of this work was conducted while the authors SK and MM were participants in the IPAM long program on Learning for Physics and the Physics of Learning. The authors thank Stefan Chmiela, Alexandre Tkatchenko, and the Tkatchenko group for providing both data and expertise.

References

Aamari, E. and Levrard, C. (2018). Stability and minimax optimality of tangential delaunay complexes for manifold reconstruction. *Discrete & Computational Geometry*, 59(4):923–971.

Brunton, S. L., Proctor, J. L., and Kutz, J. N. (2016). Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937.

Candes, E. and Tao, T. (2007). The dantzig selector: Statistical estimation when p is much larger than n. *Ann. Stat.*, 35(6):2313–2351.

Cavalli-Sforza, L. L., Menozzi, P., and Piazza, A. (1996). *The history and geography of human genes*. Princeton University Press, Princeton, N.J. (history and geography are related through PCA).

Champion, K., Lusch, B., Kutz, J. N., and Brunton, S. L. (2019). Data-driven discovery of coordinates and governing equations. *Proc. Natl. Acad. Sci. U. S. A.*, 116(45):22445–22451.

Chen, G., Little, A. V., and Maggioni, M. (2013). *Multi-Resolution Geometric Analysis for Data in High Dimensions*, pages 259–285. Birkhäuser Boston, Boston.

Chen, Y.-C. and Meilă, M. (2019). Selecting the independent coordinates of manifolds with large aspect ratios. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems 32*, pages 1086–1095. Curran Associates, Inc.

Coifman, R. R. and Lafon, S. (2006). Diffusion maps. *Applied and Computational Harmonic Analysis*, 30(1):5–30.

Das, P., Moll, M., Stamati, H., Kavraki, L., and Clementi, C. (2006). Low-dimensional, free-energy landscapes of protein-folding reactions by nonlinear dimensionality reduction. *Proceedings of the National Academy of Sciences*, 103(26):9885–9890.

Elyaderani, M. K., Jain, S., Druce, J., Gonella, S., and Haupt, J. (2017). Group-level support recovery

guarantees for group lasso estimator. pages 4366–4370.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

Haufe, S., Nikulin, V. V., Ziehe, A., Müller, K.-R., and Nolte, G. (2009). Estimating vector fields using sparse basis field expansions. In Koller, D., Schuurmans, D., Bengio, Y., and Bottou, L., editors, *Advances in Neural Information Processing Systems 21*, pages 617–624. Curran Associates, Inc.

Joncas, D., Meila, M., and McQueen, J. (2017). Improved graph laplacian via geometric Self-Consistency. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 4457–4466. Curran Associates, Inc.

Koelle, S., Zhang, H., Meilă, M., and Chen, Y.-C. (2022). Manifold coordinates with physical meaning. *Journal of Machine Learning Research*, 23.

Krenn, M., Häse, F., Nigam, A., Friederich, P., and Aspuru-Guzik, A. (2020). Self-referencing embedded strings (SELFIES): A 100% robust molecular string representation. *Mach. Learn.: Sci. Technol.*, 1(4):045024.

Lee, J. M. (2003). *Introduction to Smooth Manifolds*. Springer-Verlag New York.

Levina, E. and Bickel, P. J. (2004). Maximum likelihood estimation of intrinsic dimension. In *Advances in Neural Information Processing Systems 17 NIPS 2004, December 13-18, 2004, Vancouver, British Columbia, Canada]*, pages 777–784.

McInnes, L., Healy, J., and Melville, J. (2018a). Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

McInnes, L., Healy, J., Saul, N., and Grossberger, L. (2018b). Umap: Uniform manifold approximation and projection. *The Journal of Open Source Software*, 3(29):861.

McQueen, J., Meila, M., VanderPlas, J., and Zhang, Z. (2016). Megaman: Scalable manifold learning in python. *Journal of Machine Learning Research*, 17:148:1–148:5.

Meila, M., Koelle, S., and Zhang, H. (2018). A regression approach for explaining manifold embedding coordinates. *arXiv e-prints*, page arXiv:1811.11891.

Meurer, A., Smith, C. P., Paprocki, M., Čertík, O., Kirpichev, S. B., Rocklin, M., Kumar, A., Ivanov, S., Moore, J. K., Singh, S., Rathnayake, T., Vig, S., Granger, B. E., Muller, R. P., Bonazzi, F., Gupta, H., Vats, S., Johansson, F., Pedregosa, F., Curry, M. J., Terrel, A. R., Roučka, v., Saboo, A., Fernando, I., Kulal, S., Cimrman, R., and Scopatz, A. (2017). Sympy: symbolic computing in python. *PeerJ Computer Science*, 3:e103.

Obozinski, G., Wainwright, M. J., and Jordan, M. I. (2011). Support union recovery in high-dimensional multivariate regression. *The Annals of Statistics*, 39(1):1–47.

Rudy, S., Alla, A., Brunton, S. L., and Kutz, J. N. (2019). Data-Driven identification of parametric partial differential equations. *SIAM J. Appl. Dyn. Syst.*, 18(2):643–660.

Saul, L. K. and Roweis, S. T. (2003). Think globally, fit locally: Unsupervised learning of low dimensional manifolds. *J. Mach. Learn. Res.*, 4:119–155.

Singer, A. and Wu, H.-T. (2012). Vector diffusion maps and the connection laplacian. *Communications on Pure and Applied Mathematics*, 65(8):1067–1144.

Teh, Y. W. and Roweis, S. T. (2002). Automatic alignment of local representations. In *NIPS*.

Wainwright, M. J. (2009). Sharp thresholds for high-dimensional and noisy sparsity recovery using ℓ_1 -constrained quadratic programming (lasso). *IEEE Transactions on Information Theory*, 55:2183–2202.

Xie, T., France-Lanord, A., Wang, Y., Shao-Horn, Y., and Grossman, J. C. (2019). Graph dynamical networks for unsupervised learning of atomic scale dynamics in materials. *Nat. Commun.*, 10(1):2667.

Yalcin, I. and Amemiya, Y. (2001). Nonlinear factor analysis as a statistical method. *Statist. Sci.*, 16(3):275–294.

Ye, G.-B. and Xie, X. (2012). Learning sparse gradients for variable selection and dimension reduction. *Machine Learning*, 87(3):303–355.

Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *J. R. Stat. Soc. Series B Stat. Methodol.*

Zhang, Z. and Zha, H. (2004). Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. *SIAM J. Scientific Computing*, 26(1):313–338.

Supplementary Materials

7 Proofs

In this section we will provide proofs to the theoretical results in the main text.

7.1 Independence of Tangent Basis Selection

Proposition 7.1. *Consider alternative bases $\mathbf{T}'_i = \mathbf{T}_i\mathbf{\Gamma}_i$ where $\mathbf{\Gamma}_i$ are $d \times d$ orthonormal matrices. If $\{\mathbf{B}_i\}_{i=1}^n$ minimizes (2), then in the new tangent bases, $\{\mathbf{B}_i\mathbf{\Gamma}_i\}_{i=1}^n$ minimizes the corresponding loss function, which is constructed through replacing $\mathbf{X}_i$ by $\mathbf{\Gamma}_i\mathbf{X}_i$ in (2). Furthermore, the selected support S is independent of the basis chosen for each tangent space.*

Proof of Proposition 2. It suffices to show that the loss in (2) does not change under orthogonal transformation of individual tangent bases. As long as this holds, $\mathbf{B}_i\mathbf{\Gamma}_i$ must minimize the loss since otherwise one could argue that $J_\lambda(\mathbf{B})$ is not a minimum value for the original tangent space bases. Note that the norm $||\mathbf{B}_{\cdot j}||_2$ is unitary invariant. This is because $\mathbf{B}_{\cdot j} = (j-\text{th row of } \mathbf{B}_i)_{i=1}^n$ is constructed by stacking the $j-$th row of each $\mathbf{B}_i$. Hence the new norm is given by the norm of $(j-\text{th row of } \mathbf{B}_i\mathbf{\Gamma}_i)_{i=1}^n$; therefore the Group Lasso penalty doesn't change after changing $\mathbf{B}_i$ to $\mathbf{B}_i\mathbf{\Gamma}_i$ for each i. Finally, it holds that $||\mathbf{I}_d - \mathbf{\Gamma}_i^\top\mathbf{X}_i\mathbf{B}_i\mathbf{\Gamma}_i||_F^2 = ||\mathbf{\Gamma}_i^\top(\mathbf{I}_d - \mathbf{X}_i\mathbf{B}_i)\mathbf{\Gamma}_i||_F^2 = ||\mathbf{I}_d - \mathbf{X}_i\mathbf{B}_i||_F^2$, so the ℓ_2-loss is not changed under orthonormal transformation of the tangent bases. These rotation invariances guarantee the same support S. □

7.2 Proof of Proposition 3

We start by stating the following lemma, which gives the sufficient and necessary condition of certain matrices $\mathbf{B}_i$ to be the solution to problem (2). It also provides conditions on unique support recovery and unique solutions. The proof is standard in convex analysis literature; we follow a procedure as in (Wainwright, 2009).

Lemma 7.2. *1. Matrix $\mathbf{B}$ is the optimal solution to problem (2) if and only if there exists an matrix $Z = (z_1^\top, z_2^\top, \cdots, z_p^\top)^\top \in \mathbb{R}^{p\times nd}$ such that*

$$z_j = \begin{cases} \frac{\beta_i}{||\beta_i||} & \beta_i \neq 0 \\ \in \mathbb{R}^{nd} \text{ with } ||z_j||_2 \leq 1, & \text{otherwise} \end{cases} \tag{8}$$

and

$$\left(\mathbf{X}_1^\top(\mathbf{I}_d - \mathbf{X}_1\mathbf{B}_1), \mathbf{X}_2^\top(\mathbf{I}_d - \mathbf{X}_2\mathbf{B}_2), \cdots, \mathbf{X}_n^\top(\mathbf{I}_d - \mathbf{X}_n\mathbf{B}_n)\right) = \frac{\lambda}{\sqrt{nd}}\mathbf{Z} \tag{9}$$

.

2. If under the setting of (a), further in (8), we have $||z_i|| < 1$ whenever $\beta_i = 0$, then all optimal solutions $\widetilde{\mathbf{B}}$ of problem (2) will have support $S(\widetilde{\mathbf{B}}) \subset S(\mathbf{B})$.

3. Under setting of (a) and (b). Let $\mathbf{X}_{iS(\mathbf{B})}$ be the submatrix constructed by the $S(\mathbf{B})$ columns of of $\mathbf{X}_i$. If all $\mathbf{X}_{iS(\mathbf{B})}^\top\mathbf{X}_{iS(\mathbf{B})}$ are invertible, then the solution of problem problem (2) is unique.

Proof. Before we further explore the result, we transform the problem (2). We stack the matrices at each point together. We will now write

$$\mathbf{X} = \begin{pmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \cdots \\ \mathbf{X}_n \end{pmatrix} \in \mathbb{R}^{nd\times p}, \quad \mathbf{B} = \left(\mathbf{B}_1, \mathbf{B}_2, \cdots, \mathbf{B}_n\right) \in \mathbb{R}^{p\times nd} \tag{10}$$

Then β_j is the $j-th$ row for $\mathbf{B}$. Further let matrix

$$\mathbf{E}_i = \left(\mathbf{0}, \cdots, \mathbf{0}, \mathbf{I}_d, \mathbf{0}, \cdots, \mathbf{0}\right)^\top \in \mathbb{R}^{nd \times d} \tag{11}$$

be the block matrix with the $i-th$ block being identity matrix and the other blocks are all zeros. Then the loss function of TSLasso can be rewritten as

$$J_\lambda(\mathbf{B}) = \frac{1}{2}\sum_{i=1}^{n} ||\mathbf{E}_i^\top(\mathbf{I}_{nd} - \mathbf{X}\mathbf{B})\mathbf{E}_i||_F^2 + \frac{\lambda}{\sqrt{nd}}||\mathbf{B}||_{1,2} \tag{12}$$

where $||\mathbf{B}||_{1,2}$ is the norm defined by $\sum_{j=1}^{p}||\beta_j||_2$.

The proof of this lemma is standard technique in convex analysis. Define $h_i(\mathbf{B}) = ||\mathbf{E}_i^\top(\mathbf{I}_{nd} - \mathbf{X}\mathbf{B})\mathbf{E}_i||_F^2$ penalty part and g is the group lasso penalty.

The first step is to compute the gradient of $h_i(\mathbf{B})$ with respect to $\mathbf{B}$. For any $\mathbf{H} \in \mathbb{R}^{p\times nd}$, compute

$$h_i(\mathbf{B}+\mathbf{H}) - h_i(\mathbf{B}) \tag{13}$$

$$= \operatorname{trace}(\mathbf{E}_i^\top(\mathbf{I}_{nd} - \mathbf{X}(\mathbf{B}+\mathbf{H}))\mathbf{E}_i)^\top(\mathbf{E}_i^\top(\mathbf{I}_{nd} - \mathbf{X}(\mathbf{B}+\mathbf{H}))\mathbf{E}_i) - \operatorname{trace}(\mathbf{E}_i^\top(\mathbf{I}_{nd} - \mathbf{X}\mathbf{B})\mathbf{E}_i)^\top(\mathbf{E}_i^\top(\mathbf{I}_{nd} - \mathbf{X}\mathbf{B})\mathbf{E}_i) \tag{14}$$

$$= -2\operatorname{trace}(\mathbf{H}^\top\mathbf{X}^\top\mathbf{E}_i\mathbf{E}_i^\top(\mathbf{I}_{nd} - \mathbf{X}\mathbf{B})\mathbf{E}_i\mathbf{E}_i^\top) + O(||\mathbf{H}||_F^2) \tag{15}$$

$$= -2\left\langle \mathbf{H}, \mathbf{X}^\top\mathbf{E}_i\mathbf{E}_i^\top(\mathbf{I}_{nd} - \mathbf{X}\mathbf{B})\mathbf{E}_i\mathbf{E}_i^\top\right\rangle_F + O(||\mathbf{H}||_F^2) \tag{16}$$

Hence we can conclude that $\nabla_{\mathbf{B}} h_i(\mathbf{B}) = -2\mathbf{X}^\top\mathbf{E}_i\mathbf{E}_i^\top(\mathbf{I}_{nd} - \mathbf{X}\mathbf{B})\mathbf{E}_i\mathbf{E}_i^\top = -2\mathbf{X}^\top\mathbf{E}_i(\mathbf{I}_d - \mathbf{X}_i\mathbf{B}_i)\mathbf{E}_i^\top$, and therefore

$$\begin{aligned}\nabla_{\mathbf{B}}\frac{1}{2}\sum_{i=1}^{n}||\mathbf{I}_d - \mathbf{X}_i\mathbf{B}_i||_F^2 &= \sum_{i=1}^{n} -\mathbf{X}^\top\mathbf{E}_i(\mathbf{I}_d - \mathbf{X}_i\mathbf{B}_i)\mathbf{E}_i^\top \\ &= -\left(\mathbf{X}_1^\top(\mathbf{I}_d - \mathbf{X}_1\mathbf{B}_1), \mathbf{X}_2^\top(\mathbf{I}_d - \mathbf{X}_2\mathbf{B}_2), \cdots, \mathbf{X}_n^\top(\mathbf{I}_d - \mathbf{X}_n\mathbf{B}_n)\right).\end{aligned} \tag{17}$$

Recall that we use β_i to denote the $i-th$ row of $\mathbf{B}$. We use a similar argument in proof of lemma 2 of (Obozinski et al., 2011) and notice that the original optimization problem is convex and strictly feasible (hence strong duality holds). The primal problem is

$$\min_{\substack{\mathbf{B}\in\mathbb{R}^{p\times nd}\\ b\in\mathbb{R}^p}} \frac{1}{2}\sum_{i=1}^{n}||\mathbf{E}_i^\top(\mathbf{I}_{nd} - \mathbf{X}\mathbf{B})\mathbf{E}_i||_F^2 + \frac{\lambda}{\sqrt{nd}}\sum_{j=1}^{p} b_j \tag{18}$$

$$s.t.\ (\beta_j, b_j) \in \mathcal{K}, 1 \le j \le p \tag{19}$$

where $\mathcal{K}$ is the second-order cone as usually defined. The dual problem is given by

$$\max_{\substack{\mathbf{Z}\in\mathbb{R}^{p\times nd}\\ t\in\mathbb{R}^p}} \min_{\substack{\mathbf{B}\in\mathbb{R}^{p\times nd}\\ b\in\mathbb{R}^p}} L(\mathbf{B}, b, \mathbf{Z}, t) = \frac{1}{2}\sum_{i=1}^{n}||\mathbf{E}_i^\top(\mathbf{I}_{nd} - \mathbf{X}\mathbf{B})\mathbf{E}_i||_F^2 + \frac{\lambda}{\sqrt{nd}}\sum_{j=1}^{p} b_j + \sum_{j=1}^{p}\langle(z_j, t_j), (\beta_j, b_j)\rangle \tag{20}$$

$$s.t.\ (z_j, t_j) \in \mathcal{K}^\circ \tag{21}$$

where $z_j \in \mathbb{R}^{nd}$ is the $j-$th row of $\mathbf{Z}$. Note that $\mathcal{K}^\circ$ is the polar cone of $\mathcal{K}$ and second order cone is self-dual. Hence we have $(z_i, -\mathbf{T}_i) \in \mathcal{K}$.

Since the primal problem is strictly feasible, strong duality holds. For any pair of $(\mathbf{B}^*, b^*)$ and $(\mathbf{Z}^*, t^*)$ primal and dual solutions, they have to satisfy the KKT condtion that

$$||\beta_j^*||_2 \le b_j^*, \qquad 1 \le j \le p\,, \tag{22a}$$

$$||z_j^*||_2 \le -t_j^*, \qquad 1 \le j \le p\,, \tag{22b}$$

$$z_j^{*T}\beta_j^* + t_j^* b_j^* = 0, \qquad 1 \le j \le p\,, \tag{22c}$$

$$\nabla_B\left[\frac{1}{2}\sum_{i=1}^{n}||\mathbf{I}_d - \mathbf{X}_i\mathbf{B}_i||_F^2\right] + \mathbf{Z}^* = 0\,, \tag{22d}$$

$$\frac{\lambda}{\sqrt{nd}} + t_j^* = 0\,. \tag{22e}$$

Note that (22c)implies that $t_j^* = -\frac{\lambda}{\sqrt{nd}}$. Then by (22a) and (22b) we have $||z_j^{*T}\beta_j^*|| \leq \frac{\lambda}{\sqrt{nd}}||\beta_j||_2$. Notice that the equality holds in (22c), there fore $||z_j^*|| = \frac{\sqrt{nd}}{\lambda}$ and $b_j^* = ||\beta_j^*||$. Renormalize $z_j^* = \frac{\sqrt{nd}}{\lambda} z_j^*$ and part (a) holds. For part b, for any $j, z_j^{*T}\beta_j = ||\beta_j||_2$. Then $\beta_j = 0$ must hold for any $||z_j|| < 1$. For part (c) note that in this case the loss function is strictly convex when the original problem is restricted to minimizing over $\mathbf{B} : \beta_i = 0, \quad \forall i \notin S(\mathbf{B})$. This strict convexity implies the uniqueness of solution. □

The previous lemma provides a tool for understanding the support recovery consistency of TSLasso.

For any arbitrary $S \subset [p]$ such that $|S| = d, \operatorname{rank} \mathbf{X}_{iS} = d$ holds for all $i \in [n]$, we establish a sufficient condition on $\mathbf{X}_{iS}$ such that they can be discovered by the TSLasso. Suppose at each data point i, we decompose the matrix $\mathbf{I}_d$ by

$$\mathbf{I}_d = \mathbf{X}_{iS}\mathbf{B}_{iS}^* + \mathbf{W}_{iS} \tag{23}$$

where $\mathbf{B}_{iS}^*$s are $p \times d$ matrices that only has non zero entries in rows in S and minimizes the loss $||\mathbf{I}_d - \mathbf{X}_i\mathbf{B}_i||_F^2$. In fact, since $\mathbf{X}_{iS}$ is full rank, there exists a unique $\mathbf{B}_{iS}^* = (\mathbf{X}_{iS})^{-1}$ for each i such that $\mathbf{W}_{iS} = 0$.

The first step in proving consistency is to show a sufficient condition on $\mathbf{X}_i$ (without noisy tangent space estimation) so that the true support can be found. We first define several derived quantities of $\mathbf{X}_i$. Denoting the $j-$th column of matrix $\mathbf{X}_i$ by x_{ij}, we define

$$\text{S-incoherence} \quad \tilde{\mu}_S = \max_{i=1:n, j\in S, j'\notin S} \frac{|x_{ij}^\top x_{ij'}|}{||\nabla f_j(\xi_i)||||\nabla f_{j'}(\xi_i)||} \tag{24a}$$

$$\text{internal-colinearity} \quad \tilde{\nu}_S = \max_{i=1:n} ||(\tilde{\mathbf{X}}_{iS}^\top \tilde{\mathbf{X}}_{iS})^{-1} - \mathbf{G}_S(\xi_i)^2||. \tag{24b}$$

$$\text{maximal gradient norm} \quad \tilde{\phi}_S = \max_{i=1:n} \max_{j\in S} ||\nabla f_j(\xi_i)|| \tag{24c}$$

These are sampled version of μ_S,ν_S and ϕ_S defined on the whole manifold from (3),(4) and (6). We start with some lemmas in linear algebra.

Lemma 7.3. *Let $\mathbf{A}, \mathbf{B}$ be $d \times d$ positive definite matrices. Then $||\mathbf{A}^{-1} - \mathbf{B}^{-1}|| \leq ||\mathbf{B}^{-1}||^2||\mathbf{A} - \mathbf{B}||$*

Lemma 7.4. *Let $\mathbf{A}, \mathbf{B}$ be two $d \times d$ matrices. $\mathbf{A}$ is positive semidefinite. Denote $||\mathbf{A}||_{\infty,2}$ be the maximum ℓ_2 norm of the rows of $\mathbf{A}$. Then $||\mathbf{AB}||_{\infty,2} \leq ||\mathbf{A}||||\mathbf{B}||_F$*

Proof. Write $\mathbf{A} = (a_{ij})_{d\times d}, \mathbf{B} = (b_{ij})_{d\times d}$, then by definition

$$\begin{aligned}
||\mathbf{AB}||_{\infty,2}^2 &= \max_{i=1:d} \sum_{j=1}^d \left(\sum_{k=1}^d a_{ik} b_{kj} \right)^2 \\
&\leq \max_{i=1:d} \sum_{j=1}^d \left(\sum_{k=1}^d a_{ik}^2 \right) \left(\sum_{k=1}^d b_{kj}^2 \right) \\
&\leq \left(\max_{i=1:d} \sum_{k=1}^d a_{ik}^2 \right) \left(\sum_{j=1}^d \sum_{k=1}^d b_{kj}^2 \right) \\
&= ||\mathbf{A}||_{\infty,2}^2 ||\mathbf{B}||_F^2 \, .
\end{aligned}$$

Since $\mathbf{A}$ is positive semidefinite, we have

$$||\mathbf{A}||_{\infty,2}^2 = \max_{i=1:d} (\mathbf{AA})_{ii} \leq ||\mathbf{A}^2|| = ||\mathbf{A}||^2 \, .$$

Hence we conclude the desired result. □

Lemma 7.5. *For any arbitrary $S \subset [p]$ such that $|S| = d, \operatorname{rank} \mathbf{X}_{iS} = d$ holds for all $i \in [n]$, recall that $\delta_S = \min_{\xi\in\mathcal{M}} \min_{j\in S} ||\nabla f_j(\xi)||$, then $||(\mathbf{X}_{iS}^\top \mathbf{X}_{iS})^{-1}||_2 \leq 1 + \frac{\tilde{\nu}_S}{\delta^2}$*

Proof. Recall that $\mathbf{G}_S(\xi_i) = \text{diag}\{||\nabla f_j(\xi_i)||\}_{j,j'\in S}$ Note the upper bound

$$
\begin{aligned}
&||(\mathbf{X}_{iS}^\top \mathbf{X}_{iS})^{-1} - \mathbf{I}_d||_2 && (25)\\
=&||\mathbf{G}_S^{-1}(\xi_i)(\tilde{\mathbf{X}}_{iS}^\top \tilde{\mathbf{X}}_{iS})^{-1}\mathbf{G}_S(\xi_i)^{-1} - \mathbf{G}_S^{-1}(\xi_i)\mathbf{G}_S(\xi_i)^2\mathbf{G}_S^{-1}(\xi_i)|| && (26)\\
\leq&||(\tilde{\mathbf{X}}_{iS}^\top \tilde{\mathbf{X}}_{iS})^{-1} - \mathbf{G}_S(\xi_i)^2||||\mathbf{G}_S^{-1}(\xi_i)||^2 && (27)\\
\leq&\frac{\tilde{\nu}_S}{\delta_S^2} && (28)
\end{aligned}
$$

And the desired results come from triangle inequality. □

Also ν reflects the signal strength of the true support, which can be shown as in the following lemma.

Lemma 7.6. *Let S, δ_S be the same as in lemma 7.5. If it further holds that $\sqrt{d}\tilde{\nu}_S < \delta_S^2$ then*

$$
\min_{i=1:n}\min_{j\in S}\|[(\mathbf{X}_{iS})^{-1}]_{j\cdot}\|_2 \geq \sqrt{1 - \frac{\sqrt{d}\tilde{\nu}_S}{\delta_S^2}} \tag{29}
$$

Proof. Similar as the proof of lemma 7.5, we first upper bound the j, j' element of $(\mathbf{X}_{iS}^\top \mathbf{X}_{iS})^{-1} - \mathbf{I}_d$ by

$$
\left|\left[(\mathbf{X}_{iS}^\top \mathbf{X}_{iS})^{-1} - \mathbf{I}_d\right]_{j,j'}\right| = ||\nabla f_j(\xi_i)||^{-1}||\nabla f_{j'}(\xi_i)||^{-1}\left|\left[(\tilde{\mathbf{X}}_{iS}^\top \tilde{\mathbf{X}}_{iS})^{-1} - \mathbf{G}_S(\xi_i)^2\right]_{j,j'}\right| \leq \frac{\sqrt{d}\tilde{\nu}_S}{\delta_S^2}\ . \tag{30}
$$

Hence any diagonal element of $(\mathbf{X}_{iS}^\top \mathbf{X}_{iS})^{-1}$ is bounded by

$$
|[(\mathbf{X}_{iS}^\top \mathbf{X}_{iS})^{-1}]_{j,j'}| \geq 1 - \frac{\sqrt{d}\tilde{\nu}_S}{\delta_S^2}\ , \tag{31}
$$

leading to the desired result. □

Lemma 7.7. *Let $\{\xi_i\}_{i=1}^n$ be fixed data points on $\mathcal{M}$. Let $\tilde{\delta}_S = \min_{i=1:n}\min_{j\in S}||\nabla f_j(\xi)||$ and $\Gamma = \max_{\xi\in\mathcal{M}}\max_{j=1:p}||\nabla f_j(\xi)||$ Let $\tilde{\mu}_S, \tilde{\nu}_S, \tilde{\phi}_S$ defined from $\mathbf{X}_{iS}$ according to* (24a),(24b) *and* (24c) *respectively. Then Tangent Lasso problem* (2) *has a unique solution $\widehat{\mathbf{B}} = [\widehat{\mathbf{B}}_1, \widehat{\mathbf{B}}_2, \cdots, \widehat{\mathbf{B}}_n] \in \mathbb{R}^{p\times nd}$ with support $S(\widehat{\mathbf{B}})$ included in the true support S if $(1 + \frac{\tilde{\nu}_S}{\delta^2})\tilde{\mu}_S\tilde{\phi}_S\Gamma d < 1$. Furthermore, if $\sqrt{d}\nu_S < \delta_S^2$ and $\lambda(1 + \tilde{\nu}_S/\delta^2) < \frac{1}{2}\sqrt{n - \frac{n\sqrt{d}\nu_S}{\delta_S^2}}$, then $S(\widehat{\mathbf{B}}) = S$.*

Proof. We follow the procedure of Primal-Dual witness method (see e.g.Wainwright (2009), Obozinski et al. (2011), Elyaderani et al. (2017)).

Still considering the reformulated optimization problem (12), we first find $\widehat{\mathbf{B}}$ from minimizing a restricted optimization problem

$$
\min_{S(\mathbf{B})\subset S} J_\lambda(\mathbf{B}) = \frac{1}{2}\sum_{i=1}^n ||\mathbf{E}_i^\top(\mathbf{I}_{nd} - \mathbf{X}\mathbf{B})\mathbf{E}_i||_F^2 + \frac{\lambda}{\sqrt{nd}}||\mathbf{B}||_{1,2}. \tag{32}
$$

We then construct a dual solution $\widehat{\mathbf{Z}}$ and show that $\widehat{\mathbf{B}}$ is the solution to the original optimization problem. We write z_j as the $j-$th row of $\widehat{\mathbf{Z}}$ and decompose each $\widehat{z}_j = [\widehat{z}_{j,1}, \widehat{z}_{j,2}, \cdots, \widehat{z}_{j,n}]$. According to lemma 7.2, we can solve for $\widehat{\mathbf{Z}}$ from those optimality conditions.

First, notice that $\mathbf{B}_{iS}^* = (\mathbf{X}_{iS})^{-1}$ and

$$
\widehat{\mathbf{B}}_{iS} - \mathbf{B}_{iS}^* = -\frac{\lambda}{\sqrt{nd}}(\mathbf{X}_{iS}^\top \mathbf{X}_{iS})^{-1}\widehat{\mathbf{Z}}_{S,i}\ . \tag{33}
$$

where $\widehat{\mathbf{Z}}_S$ is constructed by concatenating the $j \in S$ row of $\widehat{\mathbf{Z}}_i$.

For an $d \times d$ matrix $\mathbf{A}$, we write $||\mathbf{A}||_{\infty,2} = \max_{i=1}^d ||a_i||_2$, where a_i is the $i-$th row of A. Then it holds that from lemma 7.4

$$
||(\mathbf{X}_{iS}^\top \mathbf{X}_{iS})^{-1}\widehat{\mathbf{Z}}_{S,i}||_{\infty,2} \leq ||(\mathbf{X}_{iS}^\top \mathbf{X}_{iS})^{-1}||||\widehat{\mathbf{Z}}_{S,i}||_F. \tag{34}
$$

Therefore recall that $||\widehat{\mathbf{Z}}_S||_{\infty,2}= 1$ we conclude that $||\widehat{\mathbf{Z}}_{S,i}||_F \leq \sqrt{d}$. And adopting lemma 7.5 we have

$$||(\mathbf{X}_{iS}^\top \mathbf{X}_{iS})^{-1}\widehat{\mathbf{Z}}_{S,i}||_{\infty,2} \leq \sqrt{d}(1+\frac{\tilde{\nu}_S}{\delta^2}) \tag{35}$$

Write $\tilde{b}_S = \sqrt{1-\frac{\sqrt{d}\nu_S}{\delta_S^2}}$. According to (33) and the assumption, $\lambda\sqrt{d}(1+\frac{\tilde{\nu}_S}{\delta^2})/\sqrt{nd} < \frac{1}{2}\tilde{b}_S$, then $||\widehat{\mathbf{B}}_{iS,j\cdot}|| \geq \frac{1}{2}\tilde{b}_S$ for each row $j \in S$.

On the other hand, for any $j' \notin S$,we have

$$\widehat{z}_{j',i} = x_{ij'}^\top \mathbf{X}_{iS}(\mathbf{X}_{iS}^\top \mathbf{X}_{iS})^{-1}\widehat{\mathbf{Z}}_{S,i}. \tag{36}$$

It suffices to verify that $||\widehat{z}_j|| < 1$ for all $j' \notin S$. For any i, we have

$$||x_{ij'}^\top \mathbf{X}_{iS}(\mathbf{X}_{iS}^\top \mathbf{X}_{iS})^{-1}||_2 \leq (1+\frac{\tilde{\nu}_S}{\delta^2})||x_{ij'}^\top \mathbf{X}_{iS}||_2 \leq \sqrt{d}(1+\frac{\tilde{\nu}_S}{\delta^2})\tilde{\mu}_S||\nabla f_{j'}(\xi_i)||\max_{j\in S}||\nabla f_j(\xi_i)|| \leq \sqrt{d}(1+\frac{\tilde{\nu}_S}{\delta^2})\tilde{\mu}_S\tilde{\phi}_S\Gamma \tag{37}$$

Directly compute that

$$\begin{aligned}
||\widehat{z}_{j'}||^2 &\leq \sum_{i=1}^n ||x_{ij'}^\top \mathbf{X}_{iS}(\mathbf{X}_{iS}^\top \mathbf{X}_{iS})^{-1}\widehat{\mathbf{Z}}_{S,i}||_2^2 \\
&\leq \sum_{i=1}^n ||x_{ij'}^\top \mathbf{X}_{iS}(\mathbf{X}_{iS}^\top \mathbf{X}_{iS})^{-1}||_2^2 ||\widehat{\mathbf{Z}}_{S,i}||_F^2 \\
&\leq d(1+\frac{\tilde{\nu}_S}{\delta^2})^2\tilde{\mu}_S^2\tilde{\phi}_S^2\Gamma^2 \sum_{i=1}^n ||\widehat{\mathbf{Z}}_{S,i}||_F^2 \\
&\leq (1+\frac{\tilde{\nu}_S}{\delta^2})^2\tilde{\mu}_S^2\tilde{\phi}_S^2\Gamma^2 d^2 < 1
\end{aligned}$$

□

This lemma is the recovery result if the tangent space is estimated without any noise. Note that this conditions also implies further results on the 'isometric' property of TSLasso. If there are two different subsets S, S' such that $|S| = |S'| = d$ and both has rank d at each data point. Then for both subsets, $\mathbf{X}_{iS}^\top \mathbf{X}_{iS}$ are invertible, and the lemma also implies that $\tilde{\mu}_S\tilde{\nu}_S d < 1$ cannot hold at the same time for both subsets. The one picked by TSLasso (usually) has a lower value of $\tilde{\nu}_S$, and will be closer to isometry to some extent.

This recovery result does not involve the tuning parameter for false inclusion. Therefore, it justifies our selection of tuning parameter that force the support has cardinality less than d. If we do observe d functions selected and they have rank d everywhere, then under incoherence condition they must be a right parameterization. To avoid false exclusion, the tuning parameter λ cannot be too large.

Now we connect these support recovery results inherent to our optimization approach with the tangent space estimation algorithm. Let $\mathbf{T}_i, \widehat{\mathbf{T}}_i$ be the orthogonal basis in $\mathbb{R}^{D\times d}$ for true and estimated tangent space respectively, and write

$$e = \max_{i=1}^n ||\mathbf{T}_i\mathbf{T}_i^\top - \widehat{\mathbf{T}}_i\widehat{\mathbf{T}}_i^\top||_2. \tag{38}$$

We have the following recovery result in the setting that gradient is estimated with some noise.

Lemma 7.8. *Let $\xi_i, i = 1:n$ be fixed data points on manifold $\mathcal{M} \subset \mathbb{R}^D$. Given S a subset of functions in dictionary $\mathcal{F} = \{f_j, j \in [p]\}$ with $|S| = d$. Suppose* $\operatorname{rank}\operatorname{grad} f_S = d$ *at each data point. Fix $\mathbf{T}_i$ as an orthonormal basis of tangent space at ξ_i, and $\widehat{\mathbf{T}}_i$ a basis for the estimated tangent space. And further define $\mathbf{X}_i = \mathbf{T}_i^\top[\nabla f_j], \widehat{\mathbf{X}}_i = \widehat{\mathbf{T}}_i^\top[\nabla f_j], j \in [p]$ where ∇ is the ambient gradient. Define $\mathbf{B}_{iS}^*, \tilde{b}_S$ the same as lemma 7.7. Define $\tilde{\mu}_S, \tilde{\nu}_S$ from* (24a) *and* (24b) *and e from* (38)*. Then let $\widehat{\mathbf{B}}$ be the solution of TSLasso problem*

$$J_\lambda(\mathbf{B}) = \frac{1}{2}\sum_{i=1}^n ||\mathbf{I}_d - \widehat{\mathbf{X}}_i\mathbf{B}_i||_F^2 + \frac{\lambda}{\sqrt{nd}}||\mathbf{B}||_{1,2}\ , \tag{39}$$

If $(1+\tilde{\nu}_S/\delta_S^2)\tilde{\mu}_S\tilde{\phi}_S\Gamma d < 1$ and $\lambda(1+\tilde{\nu}_S/\delta_S^2) < \frac{1}{2}\sqrt{n-\frac{n\sqrt{d}\nu_S}{\delta_S^2}}$, there exists a positive constant c_0 such that if $e < c_0$ then $S(\widehat{\mathbf{B}}) = S$.

Proof. The proof is direct by identifying the new $\tilde{\mu}'_S, \tilde{\nu}'_S$ parameters under noisy estimation of tangent space. The other parameters $\tilde{\phi}_S, \Gamma, \delta_S$ are not related with tangent spaces and thus remains unchanged.

Denote $\widehat{x}_i j$ the $j-$th column of $\widehat{\mathbf{X}}_i$. Similarly, to (24a), we first bound

$$\begin{aligned}\widehat{x}_{ij}^\top \widehat{x}_{ij'} &= \nabla f_j(\xi_i)^\top [\widehat{\mathbf{T}}_i \widehat{\mathbf{T}}_i^\top - \mathbf{T}_i \mathbf{T}_i^\top] \nabla f_j(\xi_i) + \nabla f_j(\xi_i)^\top \mathbf{T}_i \mathbf{T}_i^\top \nabla f_j(\xi_i) \\ &\leq ||\widehat{\mathbf{T}}_i \widehat{\mathbf{T}}_i^\top - \mathbf{T}_i \mathbf{T}_i^\top||_2 ||\nabla f_j(\xi_i)|| ||\nabla f_{j'}(\xi_i)|| + \tilde{\mu}_S ||\nabla f_j(\xi_i)|| ||\nabla f_{j'}(\xi_i)||, \quad \text{for all} \;\; j \in S, j' \notin S, i \in [n]\end{aligned}$$

So $\tilde{\mu}'_S \leq \tilde{\mu}_S + e$.

By definition, let

$$\tilde{\widehat{\mathbf{X}}}_{iS} = \left[\frac{\widehat{\mathbf{T}}_i^\top \nabla f_j(\xi_i)}{||\nabla f_j(\xi_i)||} \right]_{j \in S} = \widehat{\mathbf{X}}_{iS} \mathbf{G}(\xi_i)^{-1}$$

where $\mathbf{G}(\xi_i) = \text{diag}\{||\nabla f_j(\xi_i)||\}_{j \in S}$ and then we have

$$\tilde{\nu}'_S = ||(\tilde{\widehat{\mathbf{X}}}_{iS}^\top \tilde{\widehat{\mathbf{X}}}_{iS})^{-1} - \mathbf{G}(\xi_i)^{-2}|| \leq \tilde{\nu}_S + ||(\tilde{\widehat{\mathbf{X}}}_{iS}^\top \tilde{\widehat{\mathbf{X}}}_{iS})^{-1} - (\tilde{\mathbf{X}}_{iS}^\top \tilde{\mathbf{X}}_{iS})^{-1}||$$

It suffices to upper bound the second term. We can apply lemma 7.3, the perturbation bound of inverse of positive definite matrices. It suffice to compute

$$||(\tilde{\mathbf{X}}_{iS}^\top \tilde{\mathbf{X}}_{iS})^{-1}|| \leq ||(\tilde{\mathbf{X}}_{iS}^\top \tilde{\mathbf{X}}_{iS})^{-1} - \mathbf{G}_S(\xi_i)^2 + \mathbf{G}_S(\xi_i)^2|| \leq \tilde{\phi}_S^2 + \tilde{\nu}_S$$

And since for any $j, j' \in S$, it holds that

$$|(\tilde{\widehat{\mathbf{X}}}_{iS}^\top \tilde{\widehat{\mathbf{X}}}_{iS})_{jj'} - (\tilde{\mathbf{X}}_{iS}^\top \tilde{\mathbf{X}}_{iS})_{jj'}| \leq ||\mathbf{T}_i \mathbf{T}_i^\top - \widehat{\mathbf{T}}_i \widehat{\mathbf{T}}_i^\top|| \leq e$$

$$||\tilde{\widehat{\mathbf{X}}}_{iS}^\top \tilde{\widehat{\mathbf{X}}}_{iS} - \tilde{\mathbf{X}}_{iS}^\top \tilde{\mathbf{X}}_{iS}|| \leq ||\tilde{\widehat{\mathbf{X}}}_{iS}^\top \tilde{\widehat{\mathbf{X}}}_{iS} - \tilde{\mathbf{X}}_{iS}^\top \tilde{\mathbf{X}}_{iS}||_F \leq de$$

And thus we have

$$||(\tilde{\widehat{\mathbf{X}}}_{iS}^\top \tilde{\widehat{\mathbf{X}}}_{iS})^{-1} - (\tilde{\mathbf{X}}_{iS}^\top \tilde{\mathbf{X}}_{iS})^{-1}|| \leq (\tilde{\phi}_S^2 + \tilde{\nu}_S)^2 de$$

Hence $\tilde{\nu}'_S \leq \tilde{\nu}_S + (\tilde{\phi}_S^2 + \tilde{\nu}_S)^2 de$

For sufficiently small e, we will have $(1 + \frac{\tilde{\nu}'_S}{\delta^2})^2 \tilde{\mu}'_S \tilde{\phi}_S \Gamma d < 1$ and $\lambda(1 + \frac{\tilde{\nu}'_S}{\delta^2})^2 / \sqrt{n} < \frac{1}{2}\frac{1}{2}\sqrt{n - \frac{n\sqrt{d}\nu'_S}{\delta_S^2}}$ as these two inequality holds when $e = 0$. Then lemma 7.7 guarantees exact recovery. □

Proof of Proposition 4.3. With probability one, the following comparisons between sample based quantities and whole manifold versions holds:

$$\tilde{\mu}_S \leq \mu_S, \quad \tilde{\nu}_S \leq \nu_S, \quad \tilde{\phi}_S \leq \phi_S \tag{40}$$

Then the assumptions of the proposition guarantees that there exists a c_0 such that whenever $e < c_0$, exact recovery holds due to lemma 7.8. It suffices to notice that

$$P(S(\hat{\mathbf{B}}) = S) \leq P(e < c_0) \geq 1 - 4\left(\frac{1}{n}\right)^{\frac{2}{d}} \tag{41}$$

given by lemma 7.9. □

Lemma 7.9. *For sufficiently large constant $C > 0$, let $r_N = C(\log n/(n-1))^{1/d}$, tangent spaces $\hat{\mathbf{T}}_i$ estimated by WL-PCA in section 3.2 with neighborhood radius r_N and Gaussian kernel bandwidth $h_N \propto r_N$ satisfy that with probability at least $1 - 4(1/n)^{2/d}$*

$$\max_{i=1:n} ||\mathbf{T}_i \mathbf{T}_i^\top - \hat{\mathbf{T}}_i \hat{\mathbf{T}}_i^\top|| = O(r_N) = O((\frac{\log n}{n-1})^{\frac{1}{d}}) \,. \tag{42}$$

Proof. The proof is omitted since it is essentially the same as the proof of Proposition 15 in (Aamari and Levrard, 2018). □

Remark 7.10. Note that in this lemma, the hidden constant in big-O notation is determined by the manifold and sampling density.

8 Experiments on synthetic data – details and additional results

We include details on the synthetic experiments, as well as some additional experimental results and information in this section. In Section 8.1, we give an abstract description of the synthetic datasets we created. In Section 8.2, we look at the specific parameters and functions used to generate the data. In Section 8.3, we describe our experimental procedure. In Section 8.4, we overview our results.

8.1 Description of the synthetic manifolds and dictionaries

We generated manifolds $\mathcal{M}_i$, and dictionaries $\mathcal{F}_i$, with $i = 1, 2, 3$, using smooth and injective functions as follows. To generate $\mathcal{M}_i$, we create smooth injective function $G_i : (-1, 1)^{d_i} \to \mathbb{R}^D$, which we call *manifold functions*. Each $G_i = R_i \circ H_i$ where $R_i \in \mathbb{R}^{D\times h_i}$ is an orthonormal matrix which embeds the m_i dimensional output of H_i into $\mathbb{R}^D$, and where $H_i : (-1, 1)^d \to \mathbb{R}^{m_i}$ will consist of an affine transformation, followed by element-wise non-linear smooth functions, and ending with various non-linear combinations. Thus, $H_i = (H_1^i, \cdots, H_{m_i}^i)$ will increase the dimension of the input from d_i to m_i, while R_i will embed the m_i dimensional output of H_i into $\mathbb{R}^D$. For each $i \in \{1, 2, 3\}$, let $\mathcal{M}_i = G_i((-1, 1)^{d_i})$. For each G_i, we symbolically compute inverses $F_i = (f_1^i, \ldots, f_{d_i}^i) : \mathcal{M}_i \to (-1, 1)^{d_i}$ using Sympy (Meurer et al., 2017). As such, each $\mathcal{M}_i$ is a smooth d_i-dimensional manifold with an atlas consisting of a single chart, $\big(\mathcal{M}_i, F_i\big)$.

We create dictionaries $\mathcal{F}_i = \{f_1^i, \ldots, f_{d_i}^i, \hat{f}_1^i, \ldots, \hat{f}_{p_i'}^i\}$ of size $p_i = d_i + p_i'$ consisting of *true chart functions* $f_S^i = \{f_1^i, \ldots, f_{d_i}^i\}$ and *fake chart functions* $f_{\bar{S}}^i = \{\hat{f}_1^i, \ldots, \hat{f}_{p_i'}^i\}$, where $f^i, \hat{f}^i : \mathcal{M} \to (-1, 1)$ for all $f^i, \hat{f}^i \in \mathcal{F}_i$, and $\bar{S} = [p_i] \setminus S$. Notably, all the "fake" functions in $f_{\bar{S}}^i$ we create have significant gradient collinearity with all true functions.

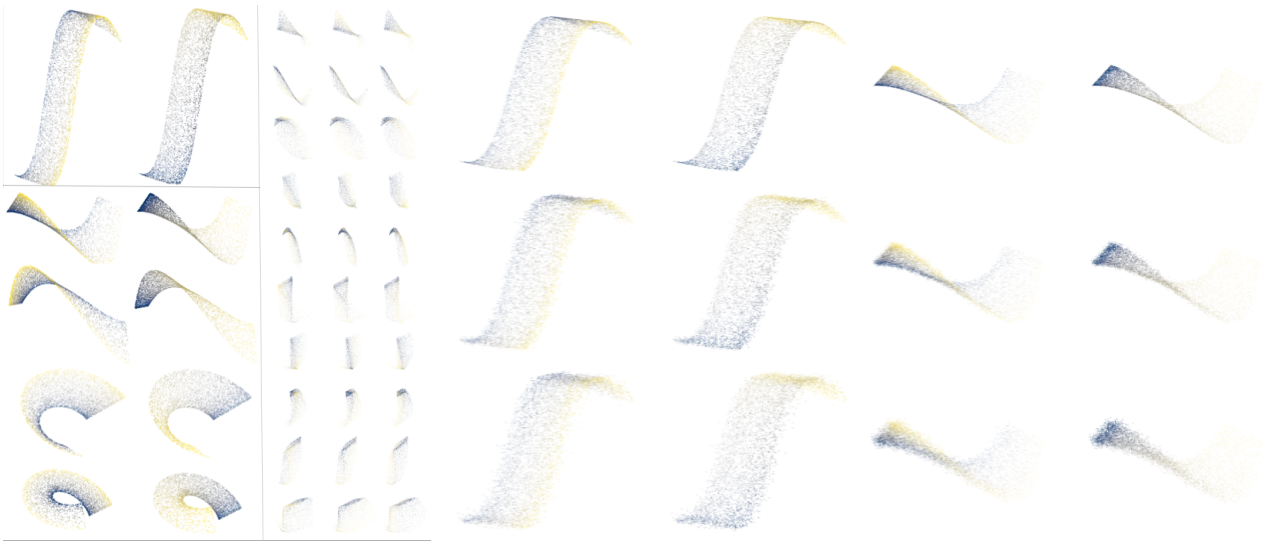
8.2 The manifolds, dictionaries, and data used in this paper

To create our synthetic data sets, we sample n points of the form $\xi = G_i(x) + \mathcal{N}(0, \sigma^2 I_D)$, where x is uniformly distributed over $(-1, 1)^{d_i}$ and where $\sigma \geq 0$ represents the standard deviation of the isotropic Gaussian noise applied in $\mathbb{R}^D$. We write $\mathcal{D}_i^\sigma$ to refer to a data set sampled as such. For our experiments, we use 10 noise levels $\sigma \in \{0.0, 0.001, 0.025, 0.05, 0.01, 0.025, 0.05, 0.1, 0.25, 0.5\}$. From our results in Figure 4, we can see that $\sigma = 0.025$ is the highest level of noise for which the recovery of TSLasso is near-perfect. In Figures 6b and 6c, we display how the manifolds we tested look at this noise level and at $\sigma = 0.05$, the lowest noise level where recovery fails. We symbolically compute the gradients $\nabla f^i, \nabla \hat{f}^i$ for all functions in $\mathcal{F}_i$ and evaluate them at each point $\xi \in \mathcal{D}_i^\sigma$. The data points thus sampled and their respective gradients represent the inputs to the TSLasso and MLasso algorithms being tested. We will refer to both the data points and gradients sampled for a *manifold function* G_i and *noise level* σ as $\mathcal{M}_i^\sigma$. Note that the algorithms *do not have access* to the original coordinates $x \in (-1, 1)^{d_i}$.

For our experiments, we use three manifold functions $G_1 : (-1, 1)^{d_1} \to \mathbb{R}^D$, $G_2 : (-1, 1)^{d_2} \to \mathbb{R}^D$, and $G_3 : (-1, 1)^{d_3} \to \mathbb{R}^D$ with $d_1 = d_2 = 2$, $d_3 = 3$, and $D = 48$. For each $G_i = R_i \circ H_i$, we sample a random orthonormal matrix $R_i \in \mathbb{R}^{D\times h_i}$, while for $H_i : (-1, 1)^{d_i} \to \mathbb{R}^{h_i}$ we use the following functions:

$$H_1(x) = \begin{bmatrix} S(t_1) \\ \cos(t_2) \\ \sin(2 \cdot t_2) \end{bmatrix} \quad H_2(x) = \begin{bmatrix} S(t_1) \\ \exp(t_2) \\ (S(t_1) + 0.5)\cos(2 \cdot \exp(t_2)) \\ (S(t_1) + 0.5)\sin(2 \cdot \exp(t_2)) \end{bmatrix} \quad H_3(x) = \begin{bmatrix} S(t_1) \\ \exp(t_2) \\ P(t_3) \\ \log(1 + S(t_1)) \cdot \sin(\exp(t_2) + P(t_3)) \\ \exp(t_2) \cdot \cos(S(t_1) + P(t_3)) \\ \cos(2 \cdot (S(t_1) + P(t_3))) \\ \sin(2 \cdot (\exp(t_2) + P(t_3))) \end{bmatrix}$$

with $m_1 = 3$, $m_2 = 4$, and $m_3 = 7$. Here $t_j(x) = a_j^T x + b_j$ represent affine functions $t_j : \mathbb{R}^{d_i} \to \mathbb{R}$, while $S(y) = \frac{1}{1+e^{-y}}$ and $P(y) = \log(1 + e^y)$ denote the sigmoid and softplus functions. The functions H_i are



(a) (b) (c)

Figure 6: **a**: Visual representations of the functions H_1(top left), H_2(bottom left), H_3(right) which we use to create the *manifold functions* $G_i = R_i \circ H_i$. Each row displays a combination of 3 coordinates of H_i, while each column j colors the points using the manifold coordinate x_j. **b, c**: Visual representation of the first three coordinates of H_1(**b**) and H_2(**c**) at different noise levels. The rows correspond to noise levels $\sigma = 0.0, 0.025, 0.05$, while each column j colors the points as a function of the manifold coordinate x_j. The second row($\sigma = 0.025$) is the highest level of noise for which the recovery of TSLasso is near-perfect, while the last row($\sigma = 0.05$) represents the lowest noise level at which all algorithms tested fail.

depicted in Figure 6a. We randomly sample a_j and b_j until we obtain manifolds which are well behaved. We consider manifolds to be well behaved when $\nu_S < 75$(see equation 4), and when $\max_{x\in(-1,1)^{d_i}} \nabla G_i(x) < 40$ and $\max_{\xi\in\mathcal{M}_i} \nabla F_i(\xi) < 40$. The restriction on ν_S ensures that the true chart functions in f_S^i don't have collinear gradients and that their gradients are reasonably close to the tangent bundle of $\mathcal{M}_i$. The restrictions on the Jacobians $\nabla G_i(x), \nabla F_i(\xi)$ ensure that the manifold is reasonably smooth and that there are no critical points of H_i and F_i on $(-1,1)^{d_i}$ and $\mathcal{M}_i$, respectively. $\nabla G_i(x)$ is estimated over a grid placed over $(-1,1)^{d_i}$, while $\nabla F_i(x)$ and ν_S are estimated over the same grid mapped by G_i into $\mathcal{M}_i$. The values of these parameters we computed for $\mathcal{M}_{1,2,3}$ are presented in Table 3.

To create the dictionaries $\mathcal{F}_i$, we use the true chart functions $f^i \in f_S^i$ to build fake chart functions of the form:

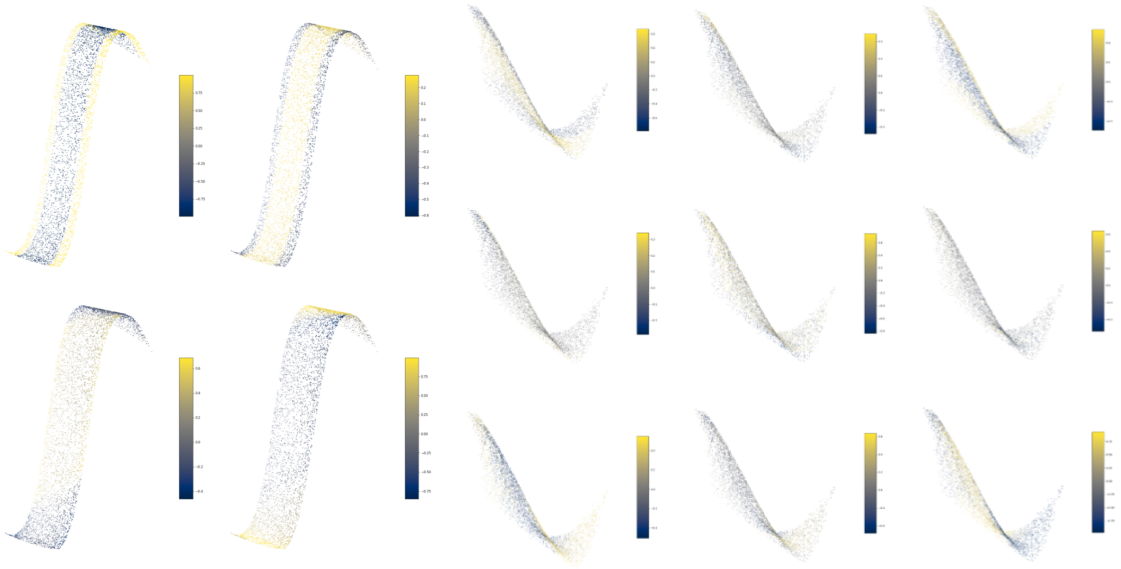
$$\hat{f}_q^i(\xi) = \xi_{j_1(q)} + \alpha_q \sin(\pi \cdot f^i_{j_2(q)}(\xi))$$

where $\xi_{j_1(q)}$ represents the $j_1(q)$-th coordinate of ξ, $f^i_{j_2(q)}$ the $j_2(q)$-th true chart function in f_S^i, $j_1(q)$ and $j_2(q)$ are randomly sampled integer indices in $[D]$ and $[d_i]$, and where α_q is sampled uniformly in $(-2,2)$. We chose the fake functions in such a way that they interact non-linearly with one of the true functions and have varying gradients over the manifolds $\mathcal{M}_i$, making recovery non-trivial. In Figure 7, we display these properties.

For a synthetic manifold $\mathcal{M}_i$ and dictionary $\mathcal{F}_i$ we can evaluate the difficulty of the recovery problem using μ_S(see equation 3) and ν_S(see equation 4). The parameter μ_S can be thought of as a renormalized incoherence between the functions in f_S^i and those in $f_{\bar{S}}^i$, while ν_S is an internal colinearity parameter. We also compute the averages of these statistics over the whole manifold. We record these values in Table 3 and note that they place us outside the theoretical recovery conditions. However, despite the difficulty of the problem, we show that TSLasso behaves robustly.

8.3 Experimental procedure

We illustrate the behavior of TSLasso on the synthetic data described in the previous section and compare its support recovery ability and run time performance against the MLasso algorithm of Koelle et al. (2022).



(a) (b)

Figure 7: Visual representations of the correlations between *true chart functions* f^i and *fake chart functions* $\hat{f}^i$. In both Figures **a** and **b**, each column k corresponds to one true chart function f^i_k for $k \in [d]$, while each column j corresponds to the fake function $\hat{f}^i_{j'} \in \hat{f}_{\bar{S}}$ with the maximum average gradient correlation with ∇f^i_j. The entry j, k in the figure is colored by $\tilde{\mathbf{X}}^\top_{\xi,\cdot j'}\tilde{\mathbf{X}}_{\xi,\cdot j}$(see Equation 3), which represents the normalized dot product of $\nabla \hat{f}^i_{j'}$ and ∇f^i_j. Figure **a** displays the output of H_1, while Figure **b** displays coordinates 3, 4, and 6 of H_3.

Sections 3 and 6 discuss the relationship between the two algorithms in some detail. Essentially, MLasso relies on embedding the data into a separate embedding space from which a basis of the tangent space corresponding to the embedding functions is pulled back into coordinate space and is used as labels for GroupLasso. TSLasso on the other hand elegantly circumvents these extra steps by using the arbitrary bases of the tangent space returned by local PCA as regression targets. Our experiments empirically show that this simplification leads to improved support recovery and run time performance. In our experiments, we use two well established embedding algorithms as subrouties for MLasso: Diffusion Maps(Coifman and Lafon, 2006) and UMAP(McInnes et al., 2018a). We use their publicly available implementations (McQueen et al., 2016) and (McInnes et al., 2018b), respectively; the two versions of MLasso are referred to as MLasso_DM and MLasso_UMAP.

We perform two experiments to compare TSLasso against MLasso_DM and MLasso_UMAP. The first experiment evaluates the support recovery of the three algorithms as a percentage of $\omega = 25$ repeated trials from different random samples I (a correct recovery occurs when the full support S is correctly identified), while the second experiment compares their run time efficiency. Each experiment will test the algorithms on a total of 3×10 synthetic data sets of the form $\mathcal{M}^\sigma_1, \mathcal{M}^\sigma_2, \mathcal{M}^\sigma_3$ generated as described above, with $\sigma \in \{0.0, 0.001, 0.0025, 0.005, 0.01, 0.025, 0.05, 0.1, 0.25, 0.5\}$. The input consists of $n = 5000$ data points in $D = 48$ dimensions and the gradients of the functions in $\mathcal{F}_i$ evaluated at these points, as well as the intrinsic dimension d and subsample size n'. In all experiments and on all $\mathcal{M}^\sigma_i$, the entire data set of size n is used for tangent space estimation. For MLasso_DM and MLasso_UMAP, it is further used to compute the embeddings of the data points. After tangent space estimation(and further steps for the MLasso algorithms), GroupLasso is applied to a uniformly random subset of size n' of the whole dataset, with this process being independently repeated ω times. The tangent space estimation(and additional computations for the MLasso algorithms) plus the ω repeats of GroupLasso constitute a full run of an algorithm. For each data set $\mathcal{M}^\sigma_i$ we perform $\omega = 25$ replications for the support recovery tests. We do this because, from preliminary experiments, we found that at the given n the variance in the sample w.r.t. the manifold estimation is negligible and need not be repeated.

For the run time tests we perform $\omega = 10$ runs, each from a new $\mathcal{D}^\sigma_i$. The local tangent space kernel bandwidth ϵ is estimated using Joncas et al. (2017). For MLasso_UMAP we use 200 neighbors and a minimum distance of 0.

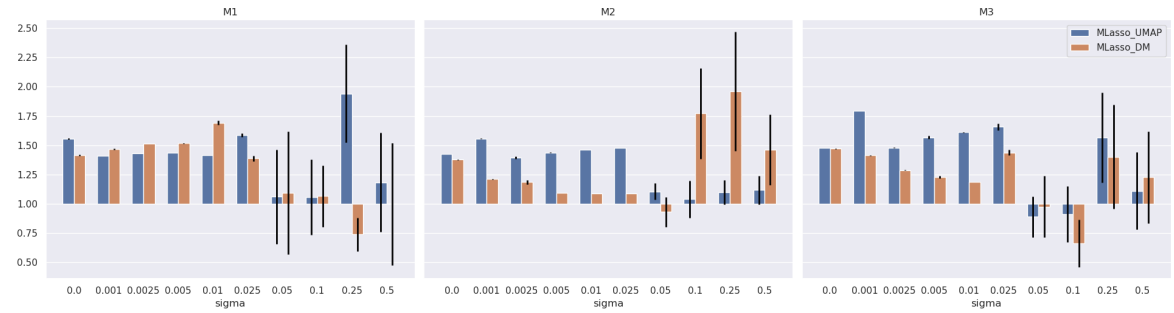
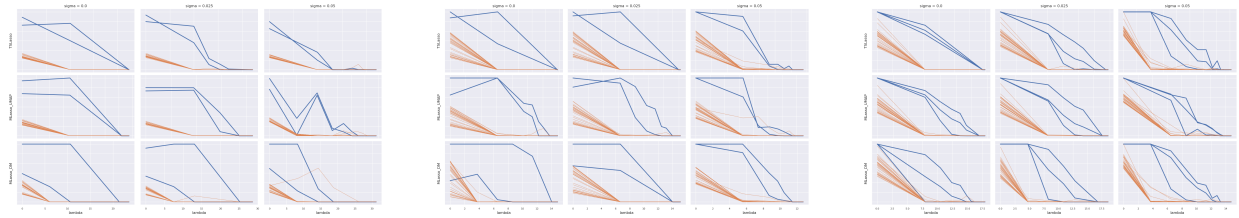


Figure 8: Speedup of TSLasso relative to MLasso_DM and MLasso_UMAP averaged over 10 independent trials. The error bars represent the standard deviation. We observe that TSLasso achieves around 1.25 - 1.5 speed up over the other algorithms. This is due to TSLasso not requiring an embedding algorithm or other extra steps that the MLasso algorithms do. The results for $\sigma \geq 0.05$ are inconsistent because at these noise levels the search for λ takes a variable number of steps and doesn't always terminate within the allowed number of iterations.



(a) (b) (c)

Figure 9: The regularization paths obtained by the three algorithms on $\mathcal{M}_1(\mathbf{a})$, $\mathcal{M}_2(\mathbf{b})$, $\mathcal{M}_3(\mathbf{c})$, averaged over 25 independent trials. An orange line(a *fake chart function*) crossing on top of a blue line(a *true chart function*) indicates a consistent failure to fully recover the support. Note that for $\sigma \leq 0.025$, the binary search for λ ends after just 1 step for TSLasso and MLasso_DM, and after about 6 steps for MLasso_UMAP, in failure.

Parameters are summarized in Table 3.

Dataset	n	d	D	p	n'	ω	μ_S	ν_S	avg μ_S	avg ν_S	max ∇G_i	max ∇F_i
$\mathcal{M}_1$	5000	2	48	38	500	25 or 10	0.99	33.91	0.43	6.55	12.95	5.83
$\mathcal{M}_2$	5000	2	48	38	500	25 or 10	0.55	46.39	0.17	8.18	20.82	19.11
$\mathcal{M}_3$	5000	3	48	39	500	25 or 10	0.97	51.50	0.26	12.37	14.92	9.81

Table 3: Parameters used for the three synthetic experiments.

8.4 Detailed experimental results

The results are summarized in Figures 4, 8, and 9, as well as in Tables 4a and 4b. Our experiments empirically show that TSLasso achieves improved support recovery, especially on the $\mathcal{M}_3^\sigma$ datasets(see Figure 4), and around 1.25-1.5 speedup with respect to MLasso_UMAP and MLasso_DM (see Figure 8). While the latter is a direct cause of removing the embedding, Riemannian metric estimation, and pull back steps of MLasso, we posit that the former is a result of eliminating the possibility of the embedding algorithm producing imperfect targets for the regression problem. This claim is supported by the near perfect recovery achieved by TSLasso on all datasets with noise levels up to $\sigma = 0.025$, which contrasts with the inconsistent results of the other two algorithms. In particular, we note the higher frequency of failure of MLasso_DM on all of the manifolds. We believe that this behavior stems from the manifolds' aspect ratio, which causes the Diffusion Maps algorithm to create rank deficient mappings (Chen and Meilă, 2019). Furthermore, we also note the relatively low performance of MLasso_UMAP on the $\mathcal{M}_3^\sigma$ datasets. Finally, we note that beyond $\sigma = 0.05$ all algorithms fail to recover the

true support which is expected due to the high noise levels as shown in Figures 6b and 6c.

9 Experiments with real MDS data

We include details on the MDS data preprocessing, our experiments, as well as some additional experimental results and information in this section. The settings on these data are shown in table 5 (same values as in the main paper).

9.1 Preprocessing the MDS data

MDS data are generated originally in $3 \times N_a$ coordinates; these require preprocessing to ensure the invariance to translation and rotation before neighbors can be computed. For this, we follow the same procedure as Koelle et al. (2022), using the code available at `https://github.com/sjkoelle/montlake`, which we briefly describe here for completeness.

One first obtains an Euclidean group-invariant featurization of the atomic coordinates as a vector of planar angles $a_i \in \mathbb{R}^{3\binom{N_a}{3}}$: the planar angles formed by triplets of atoms in the molecule. One then performs an SVD on this featurization, and project the data onto the top $D = 50$ singular vectors to remove linear redundancies. Note that this represents a particular metric on the molecular *shape space*.

The dictionaries we considered are constructed on *bond diagram*, a priori information about molecular structure garnered from historical work. Building a dictionary based on this structure is akin to many other methods in the field (Krenn et al., 2020; Xie et al., 2019). Specifically, this dictionary consist of all equivalence classes of 4-tuples of atoms implicitly defined along the molecule skeletons.

Since original angular data featurization is an overparameterization of the shape space, one cannot use automatically obtained gradients in TSLasso. They first must be projected on the tangent bundle of the *shape space*[6] as it is embedded in $\mathbb{R}^D$.

9.2 Additional details on the MDS experimental results

We plot the incoherence for Ethanol and Malonaldehyde as the heatmap in figure 10b and 10f, which present two groups of highly dependent torsions, corresponding to the two bonds between heavy atoms in the molecules. For example, the rotation angle (torsion) marked in orange around axis C(1)–C(2) can be measured w.r.t. any one of {O(3),H(4),H(5)} atoms and any of {H(6),H(7),H(8)} atoms, for a total of 9 almost equivalent torsions.

Therefore, the success of a recovery algorithm under this high indeterminacy is to select a pair of incoherent torsions out of these dictionaries. Figures 10h and 10d show support recovery frequencies for sets of size $d = s = 2$ using TSLasso on ethanol and malonaldehyde data respectively. As we desired, TSLasso selects one function from each of the two groups of torsions in most replicates; e.g., for ethanol, always a rotation with axis C(1)–C(2), and one with axis C(1)–O(3) are chosen. Thus, the TSLasso algorithm is robust and performs soundly even under these conditions of high coherence.

9.2.1 Visualisation of recovered coordinates

Here we display the coordinate functions selected by TSLasso overlayed over the Diffusion maps embeddings of Ethanol, in Figures 11a and 11b, and and Malonaldehyde, in Figures 12a and 12b.

For Ethanol, it is easy to see that the two functions selected from the TSLasso indeed parametrize the structure of the data, as each function is varying along one of the two circles generating the torus. However, in the case of the Malonaldehyde data, while the embedding is topologically a torus, it is much harder to select coordinate functions from $\mathcal{F}$ by visual inspection alone. For illustration, we also visualize a torsion which is not a coordinate function.

[6] The shape space is the manifold of equivalence classes of $N_a \times 3$ atomic coordinates w.r.t. the invariance group considered. The manifolds that are approximated by the MDS data in these experiments are *submanifolds* of the shape space.

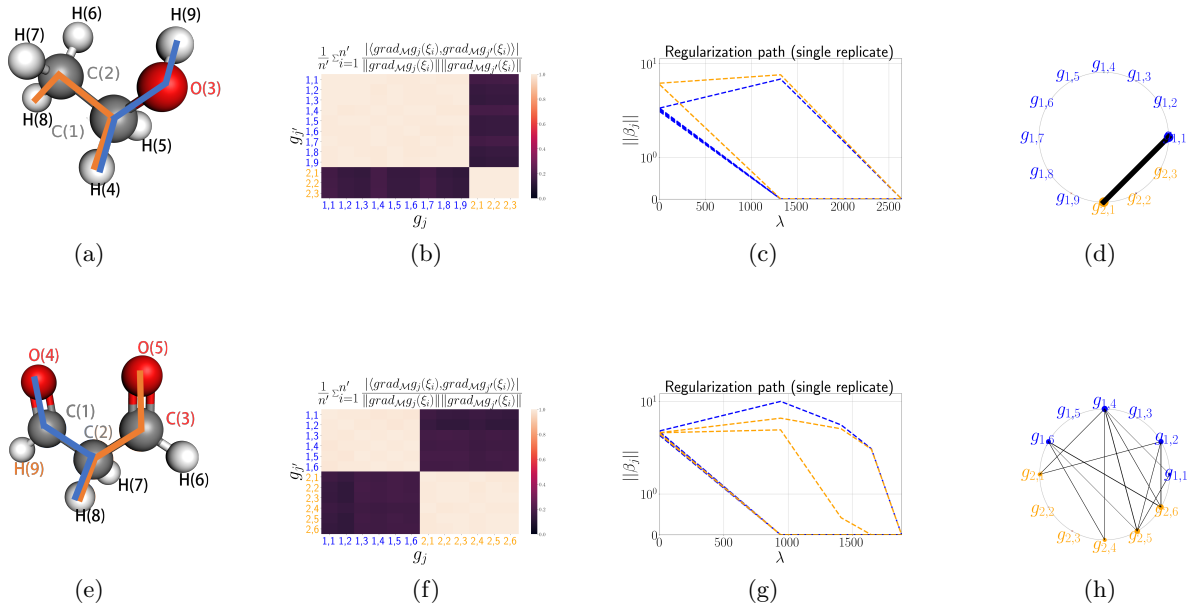
	Algo.	TSLasso		MLasso_UMAP		MLasso_DM	
Manifold	σ	mean	std	mean	std	mean	std
$\mathcal{M}_1$	0.00	**1.0000**	0.0000	**1.0000**	0.0000	0.9600	0.1960
	0.001	**1.0000**	0.0000	**1.0000**	0.0000	0.4400	0.4964
	0.0025	**1.0000**	0.0000	**1.0000**	0.0000	0.3600	0.4800
	0.005	**1.0000**	0.0000	**1.0000**	0.0000	0.2400	0.4271
	0.01	**1.0000**	0.0000	**1.0000**	0.0000	0.4000	0.4899
	0.025	0.8400	0.3666	**1.0000**	0.0000	0.5200	0.4996
	0.05	**0.5600**	0.4964	0.5200	0.4996	0.0000	0.0000
	0.1	**0.3200**	0.4665	0.2800	0.4490	0.1600	0.3666
	0.25	**0.0400**	0.1960	0.0000	0.0000	0.0800	0.2713
	0.5	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
$\mathcal{M}_2$	0.0	**1.0000**	0.0000	0.9200	0.2713	0.5200	0.4996
	0.001	**1.0000**	0.0000	0.9600	0.1960	0.7200	0.4490
	0.0025	**1.0000**	0.0000	0.8400	0.3666	0.9600	0.1960
	0.005	**1.0000**	0.0000	**1.0000**	0.0000	0.9200	0.2713
	0.01	**1.0000**	0.0000	0.9200	0.2713	0.9600	0.1960
	0.025	**1.0000**	0.0000	0.8000	0.4000	1.0000	0.0000
	0.05	**0.4400**	0.4964	**0.4400**	0.4964	0.4400	0.4964
	0.1	**0.1600**	0.3666	0.0000	0.0000	0.0000	0.0000
	0.25	**0.2000**	0.4000	0.1200	0.3250	0.0400	0.1960
	0.5	**0.3200**	0.4665	0.0800	0.2713	0.0400	0.1960
$\mathcal{M}_3$	0.0	**1.0000**	0.0000	0.6000	0.4899	0.0000	0.0000
	0.001	**1.0000**	0.0000	0.8000	0.4000	0.0000	0.0000
	0.0025	**1.0000**	0.0000	0.5200	0.4996	0.0000	0.0000
	0.005	**1.0000**	0.0000	0.8000	0.4000	0.0000	0.0000
	0.01	**1.0000**	0.0000	0.9200	0.2713	0.0400	0.1960
	0.025	**0.9200**	0.2713	0.8800	0.3250	0.2400	0.4271
	0.05	0.2400	0.4271	**0.3200**	0.4665	0.1600	0.3666
	0.1	**0.0400**	0.1960	**0.0400**	0.1960	0.0000	0.0000
	0.25	**0.0800**	0.2713	0.0000	0.0000	0.0400	0.1960
	0.5	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

(a)

	Algo.	MLasso_UMAP		MLasso_DM	
Manifold	σ	mean	std	mean	std
$\mathcal{M}_1$	0.00	**1.5546**	0.0058	**1.4158**	0.0053
	0.001	**1.4116**	0.0002	**1.4695**	0.0053
	0.0025	**1.4330**	0.0001	**1.5155**	0.0011
	0.005	**1.4365**	0.0001	**1.5169**	0.0006
	0.01	**1.4150**	0.0006	**1.6916**	0.0175
	0.025	**1.5859**	0.0155	**1.3873**	0.0232
	0.05	**1.0605**	0.4028	**1.0936**	0.5264
	0.1	**1.0565**	0.3203	**1.0649**	0.2605
	0.25	**1.9410**	0.4179	0.7376	0.1446
	0.5	**1.1829**	0.4225	0.9984	0.5234
$\mathcal{M}_2$	0.00	**1.4271**	0.0001	**1.3771**	0.0027
	0.001	**1.5571**	0.0021	**1.2117**	0.0024
	0.0025	**1.3939**	0.0099	**1.1849**	0.0190
	0.005	**1.4384**	0.0004	**1.0933**	0.0000
	0.01	**1.4600**	0.0003	**1.0858**	0.0000
	0.025	**1.4788**	0.0013	**1.0898**	0.0001
	0.05	**1.1059**	0.0693	0.9300	0.1270
	0.1	**1.0388**	0.1574	**1.7726**	0.3882
	0.25	**1.0980**	0.1017	**1.962**	0.5087
	0.5	**1.1183**	0.1217	**1.4625**	0.3013
$\mathcal{M}_3$	0.00	**1.4769**	0.0009	**1.4699**	0.0038
	0.001	**1.7949**	0.0013	**1.4132**	0.0016
	0.0025	**1.4798**	0.0006	**1.2866**	0.0017
	0.005	**1.5663**	0.0128	**1.2282**	0.0096
	0.01	**1.6118**	0.0028	**1.1880**	0.0012
	0.025	**1.6571**	0.0265	**1.4387**	0.0215
	0.05	0.8887	0.1743	0.9747	0.2632
	0.1	0.9090	0.2390	0.6623	0.2017
	0.25	**1.5656**	0.3850	**1.3999**	0.4443
	0.5	**1.1101**	0.3288	**1.2258**	0.3945

(b)

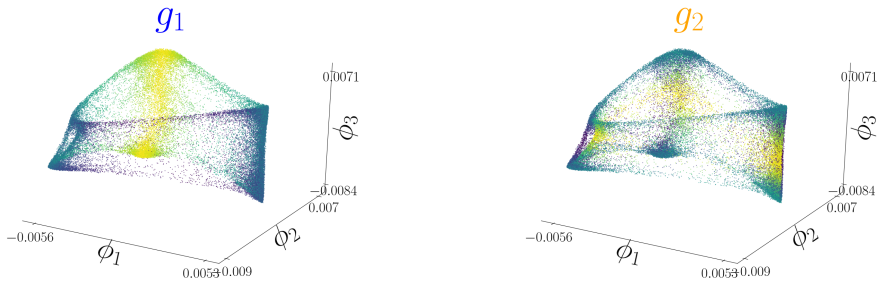
Table 4: **a**: Successful support recoveries as a proportion of 25 independent trials for TSLasso, MLasso_DM, and MLasso_UMAP. Our algorithm consistently achieves near perfect recovery on all manifolds for noise levels up to $\sigma = 0.025$, while the performance of MLasso_DM and MLasso_UMAP drops, particularly on the $\mathcal{M}_3$ datasets. **b**: Speedup of TSLasso relative to MLasso_DM and MLasso_UMAP averaged over 10 independent trials. We observe that TSLasso achieves around 1.25 - 1.5 speed up over the other algorithms due to its relative simplicity.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 10: Results from molecular dynamics data. 10a, 10e show bond diagrams for ethanol and malonaldehyde, respectively. 10b and 10f show the heatmap of cosines (incoherences) of dictionary functions. The color is lighter when there is more colinearity (note that in thise figures the collinearities are either nearly 1 or nearly 0). The dictionaries are $\mathcal{F} = \{g_{1,1:9}, g_{2,1:3}\}$ for ethanol, and $\mathcal{F} = \{g_{1,1:6}, g_{2,1:6}\}$ for malonaldehyde. The grouping refers to the common axis for the torsions in a group; for instance the group $g_{1,1:6}$ denotes the six torsions associated with the blue C(1)–C(2) axis in malonaldehyde. 10c, 10g are regularization paths for a single replicate of ethanol and malonaldehyde. Note that in both figures there are a redundant trajectory of two functions that are added together. 10d, 10h Selection of pairs of functions for ethanol and malonaldehyde over replicants using TSLasso. The node point on the circles represents all functions in the dictionary and the number along the lines are frequencies of each pairs selected over 25 repetitions. 10d means in all 25 repetitions, TSLasso selects $g_{1,1}$ and $g_{2,1}$, which are the bond torsions around C-C bond and C-O bond respectively. 10h show that in 24 out of 25 replicates, TSLasso is able to select one function from each highly colinear function group.

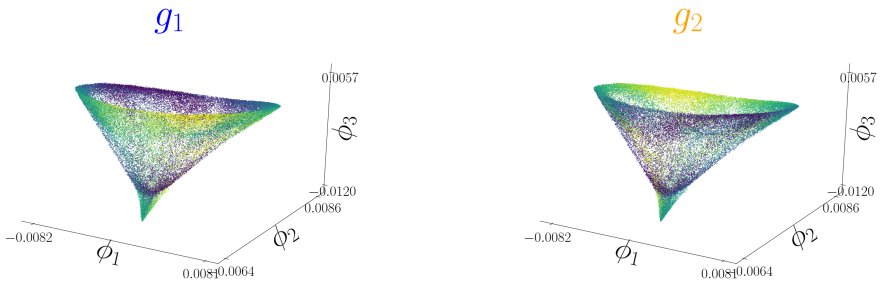


(a) (b)

Figure 11: Diffusion map embedding for ethanol MDS data. Data points are colored by the two torsions found by TSLasso, denoted here generically as g_1, g_2.

Dataset	n	d	D	p	n'	ω	ϵ	N_a
Eth	50000	2	50	12	100	25	3.5	9
Mal	50000	2	50	12	100	25	3.5	9
Tol	50000	1	50	30	100	25	1.9	15

Table 5: Parameters for the Molecular Dynamics Simulation data experiments: Eth(Ethanol), Mal(Malonaldehyde) and Tol(Toluene), N_a number of atoms.



(a) (b)

Figure 12: Diffusion map embedding for Malonaldehyde data. Data points are colored by the two torsions found by TSLasso, denoted here generically as g_1, g_2.

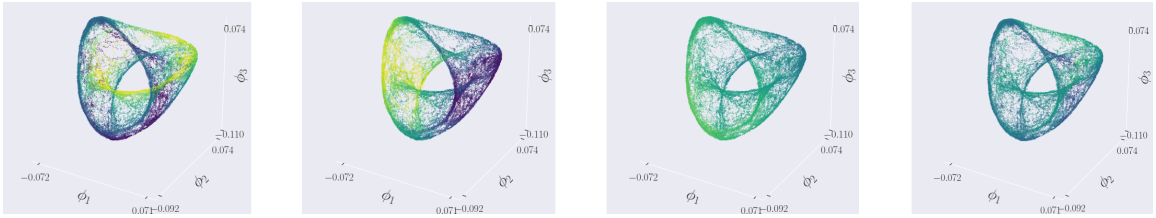


Figure 13: Diffusion Maps embedding for another molecule, Dimethilfuran, MDS data. The embedding is colored, from left to right, by two torsions which are the correct coordinate functions for this molecule; a third torsion which has high variation normal to the manifold (hence only small variation along $\mathcal{M}$; and a fourth function that cannot be coordinate function because it oscillates on $\mathcal{M}$.