

Approximation of Functions on Manifolds in High Dimension from Noisy Scattered Data[†]

Shira Faigenbaum-Golovin, David Levin

Abstract

In this paper, we consider the fundamental problem of approximation of functions on a low-dimensional manifold embedded in a high-dimensional space. Classical approximation methods, developed for the low-dimensional case, are challenged by the high-dimensional data, and the presence of noise. Here, we introduce a new approximation method that is parametrization free, can handle noise and outliers in both the scattered data and function values and does not require any assumptions on the scattered data geometry. Given a noisy point-cloud situated near a low dimensional manifold and the corresponding noisy function values, the proposed solution finds a noise-free, quasi-uniform manifold reconstruction as well as the denoised function values at these points. Next, this data is used to approximate the function at new points near the manifold. We prove that in the case of noise-free samples the approximation order is $O(h^2)$, where h depends on the local density of the dataset (i.e., the fill distance), and the function variation. We demonstrate the effectiveness of our approach by examining smooth and non-smooth functions on different manifold topologies, under various noise levels.

Keywords: Approximation of functions, High dimensions, Scattered Data, Noisy data, Manifold learning, Dimensional Reduction .

MSC: Primary 41-xx, 65Dxx; Secondary 41A63, 65D99.

§1. Introduction

Function approximation is a fundamental problem, with foundations laid for the low-dimensional case. Given scattered data and function values, we wish to use the regularity within the data and the function values, for the approximation of the function at a new point. With the dissemination of high-dimensional data, the interest in this question was renewed, and the task was rephrased to deal with the new challenges that this data raise. While for the approximation task for low dimensional data various

[†]Shira is grateful to the Eric and Wendy Schmidt Fund for Strategic Innovation, the Zuckerman-CHE STEM Program, Duke University, and in part the Simons Foundation under Grant Math+X 400837 for supporting her research.

methods were suggested to solve this problem (e.g., splines, Radial Basis Functions (RBF), or Moving Least-Squares [21]), in high dimensions, approximating a function is still a non-trivial task. High-dimensional data pose many new challenges due to the curse of dimensionality, the presence of noise, and the unknown topology of the scattered data. For example, in the case of a smooth function, the approximation rates deteriorate severely with the growth of the dimension, the reason being that the amount of sampled data should grow exponentially with respect to the dimension if one wishes to maintain the same approximation order.

In this paper, we study the problem of approximating functions on manifolds in a high-dimensional space in a noisy scenario. Let \mathcal{M} be a d -dimensional manifold in \mathbb{R}^n , where $d \ll n$, and let $X = \{x_i\}_{i=1}^K \subset \mathbb{R}^n$, be a set of points sampled from \mathcal{M} . Also let $f : \mathcal{M} \subset \mathbb{R}^n \rightarrow \mathbb{R}^s$ be a function, along with its values $\{f(x_i)\}_{i=1}^K$ at the given points. In the current setting, noise may be present both in the domain of the function and in its co-domain. We are looking for a noise-free approximation to f at a new point near \mathcal{M} .

The existing approximation methods for approximating functions in high dimensions, fall broadly into two categories: the domain of the function is a smooth manifold or not. In case the data does not lay on a manifold, several methods were suggested. For example, solutions that treat non-smooth multivariate functions, [1], are based on sparse occupancy trees [5], Radial Basis Functions [15] (discussed in detail below), or address the problem in the case where the values of the function lie on a manifold [19].

On the other hand, if the high-dimensional data resides on a low-dimensional manifold, this information can be exploited to improve the approximation via one of the following two approaches: approximating in low dimension after dimension reduction, or alternatively approximating in high dimension. At times, reducing the dimension (e.g., in PCA [26], Multidimensional Scaling [12], Linear Discriminant Analysis [18], Locality Preserving Projections [20], Locally Linear Embedding [27], ISOMAP [30], Diffusion Maps [10], and Neural Networks in their general form, [24]) can lead to a better approximation (in terms of handling the challenge of the dimensionality, as well as the noise in the data). However, it may be non-efficient if the data volume is very large, and in addition, may result in information loss (due to some assumptions that need to be made on the data, e.g., regarding the data geometry, or the intrinsic dimension).

In contrast, using the manifold assumption, the approximation can be performed in high dimensions. For example, approximation of functions on manifolds is studied using local polynomials [4], wavelets [11], local linear regression [4], or neural networks [2, 28, 9]. In [6] the authors propose to deal with noise and outliers by a method inspired by the k -nearest neighbor's regression and the local median filtering, while the outliers are filtered using a distance to a measure function. For smooth functions on $[0, 1]^N$ which depend on a much smaller number ℓ of variables, a solution was suggested in [14]. In addition, a recent paper, [29], proposed a solution based on Moving Least-Squares (MLS) that was designed to deal with noisy data with good rates of approximation.

In this paper, we propose a method of approximation of functions that extends the Manifold Locally Optimal Projection (MLOP) algorithm [17] for denoising the data, and uses its advantages to boost the method of Radial Basis Functions (RBF) [15, 7]. We introduce this duet for approximation in high dimensions under noisy conditions

(both in the domain and in the co-domain of the function). The new method name is Function approximation and Manifold Locally Optimal Projection (FMLOP). While the RBF approximation alone is sensitive to noise and non-uniform data, as will be shown in this paper, the proposed solution is robust under these scenarios.

The rest of this paper is organized as follows: we start by summarizing our main contribution; in Section 2 we provide the theoretical framework of the MLOP and RBF methods; Section 3 describes the proposed methodology for approximating functions in high-dimension, along with a theoretical analysis of the method. Numerical examples that demonstrate the effectiveness of the proposed solution are listed in Section 4.

Main Contribution

The main contribution of this paper is to provide a simple and efficient method for function approximation from noisy data in high-dimension. Commonly, the presence of noise challenges the existing approximation methods. Primly, due to the lack of additional information related to the noise model, its magnitude, or its location. The solution reported herein deals with high amounts of noise and outliers by cleaning the data (both the scattered data and the function values). The power of the proposed methodology is that we assume that high-dimensional input data lies on an intrinsically low-dimensional Riemannian manifold and use the intrinsic connections between different samples for the denoising task. This step not only denoises the existing data but also paves the way to improving the approximation order of function approximation at new points. In addition, since the pre-processing step is performed only once per problem setting, the function approximation at new points is very efficient.

Moreover, the proposed method demonstrates that, naturally, denoising the data before performing approximation is beneficial. As will be seen below, the MLOP and RBF duet outperforms the classical RBF function approximation. The analysis performed in this paper demonstrates the effectiveness of the data-denoising method prior to the approximation task.

§2. Preliminaries

2.1. The MLOP framework

The *Manifold Locally Optimal Projection* (MLOP) [17] was designed to reconstruct the underlying manifold while dealing with noisy data in high-dimensions. The MLOP enhances the Locally Optimal Projection (LOP) method initially introduced in [25], designed to approximate two-dimensional surfaces in \mathbb{R}^3 from point-set data. The MLOP algorithm is simple, fast, and efficient, and does not require any additional assumptions. The theoretical analysis, accompanied by numerical examples in [17] show that the methodology is robust in case of noisy scenarios in high-dimension. Herein, we provide a concise overview of the MLOP method and its key properties that later are used for approximation of functions.

The setting of the high-dimensional reconstruction problem is the following: Let \mathcal{M} be a manifold in \mathbb{R}^n of unknown intrinsic dimension $d \ll n$. Given a noisy point-cloud

$P = \{p_j\}_{j=1}^J \subset \mathbb{R}^n$ situated near the manifold \mathcal{M} such that P is a h - ρ set, we wish to find a new point-set $Q = \{q_i\}_{i=1}^I \subset \mathbb{R}^n$ which will serve as a noise-free approximation of \mathcal{M} . First, let us introduce the h - ρ condition, defined for scattered-data approximation of functions (which is an adaptation of the condition in [21] for low-dimensional data), to handle finite discrete data on manifolds:

Definition 2.1. h - ρ sets of fill distance h and density $\leq \rho$ with respect to a manifold \mathcal{M} . Let \mathcal{M} be a manifold in \mathbb{R}^n and consider sets of data points sampled from \mathcal{M} . We say that such a set $P = \{p_j\}_{j=1}^J$ is an h - ρ set if:

1. h is the fill distance, i.e., $h = \max_{y \in \mathcal{M}} \min_{p_j \in P} \|y - p_j\|$.
2. $\#\{P \cap \bar{B}(y, kh)\} \leq \rho k^n$, $k \geq 1$, $y \in \mathcal{M}$.

Here $\#Y$ denotes the number of elements in a set Y and $\bar{B}(x, r)$ denotes the closed ball of radius r centered at x .

In the manifold reconstruction scenario, we seek for a solution in the form of a new, quasi-uniformly distributed point-set Q that will replace the given data P and provide a noise-free approximation of \mathcal{M} . This is achieved by leveraging the well-studied weighted L_1 -median [31] used in the LOP algorithm and requiring a quasi-uniform distribution of points $q_i \in Q$. These ideas are encoded by the cost function

$$G(Q) = E_1(P, Q) + \Lambda E_2(Q) = \sum_{q_i \in Q} \sum_{p_j \in P} \|q_i - p_j\|_{H_\epsilon} w_{i,j} + \sum_{q_i \in Q} \lambda_i \sum_{q_{i'} \in Q \setminus \{q_i\}} \eta(\|q_i - q_{i'}\|) \hat{w}_{i,i'}, \quad (2.1)$$

where the weights $w_{i,j}$ are rapidly decreasing smooth functions, where $w_{i,j} = \exp\{-\|q_i - p_j\|^2/h_1^2\}$ and $\hat{w}_{i,i'} = \exp\{-\|q_i - q_{i'}\|^2/h_2^2\}$. The L_1 -norm used in [25] is replaced by the “norm” $\|\cdot\|_{H_\epsilon}$ introduced in [22] as $\|v\|_{H_\epsilon} = \sqrt{v^2 + \epsilon}$, where $\epsilon > 0$ is a fixed parameter (in our case we take $\epsilon = 0.1$). The parameters h_1 and h_2 are selected to guarantee a sufficient amount of P or Q points for the reconstruction (for more details, see the subsection “Optimal Neighborhood Selection” below). Also, $\eta(r)$ is a decreasing function such that $\eta(0) = \infty$; in our case we take $\eta(r) = \frac{1}{3r^3}$. Finally, $\{\lambda_i\}_{i=1}^I$ are constant balancing parameters.

In order to solve the problem with the cost function (2.1), we look for a point-set Q that minimizes $G(Q)$. The solution Q is found via the gradient descent iterations

$$q_{i'}^{(k+1)} = q_{i'}^{(k)} - \gamma_k \nabla G(q_{i'}^{(k)}), \quad i' = 1, \dots, I, \quad (2.2)$$

where the initial guess $\{q_i^{(0)}\}_{i=1}^I = Q^{(0)}$ consists of points sampled from P , and $\nabla G(q_{i'}^{(k)})$ is the gradient of G as given in equation 2.1. For the explicit details about gradient calculation as well as the selection of γ_k and λ_i parameters see the Appendix 5. The code for the MLOP algorithm, along with examples, can be found in <https://github.com/shirafai-gen/MLOP>

Norm calculation in a noisy scenario. The reasoning in terms of Euclidean distances, which is the cornerstone of the MLOP method, works well in low dimensions, e.g., for the reconstruction of surfaces in 3D, but breaks down in high dimensions once the noise is present. To deal with this issue, a dimension reduction is performed via random linear sketching [32]. It should be emphasized that the dimension reduction

procedure is utilized solely for the calculation of norms, and the manifold reconstruction is performed in the high-dimensional space. Thus, given a point $x \in \mathbb{R}^n$, we project it to a lower dimension $m \ll n$ using a random matrix, S , with certain properties. Subsequently, the norm of $\|S^t x\|$ will approximate $\|x\|$. The construction of S is carried out in the following steps: (a) sample $G \in \mathbb{R}^{J \times m}$ with $G \sim N(0, 1)$; (b) compute $B \in \mathbb{R}^{n \times m}$ as $B := P^t G$; (c) calculate the QR decomposition of B as $B = SR$, and use S as the dimension reduction matrix.

Optimal neighborhood selection. The parameters h_1 and h_2 used in the functions $w_{i,j} = \exp\{-\|q_i - p_j\|^2/h_1^2\}$ and $\hat{w}_{i,i'} = \exp\{-\|q_i - q_{i'}\|^2/h_2^2\}$, respectively, are closely related to the fill distance of the P -points and the Q -points. Due to the importance of the optimal selection of these parameters, we quote here several definitions and results from [17]. First, we adapt the fill-distance notion defined for scattered-data approximation functions (in [21], defined for low-dimensional data), to handle finite discrete data on manifolds. Next, we define the \hat{h}_0 parameter which will guarantee a sufficient amount of active points in the neighborhood of each q_i point.

Definition 2.2. Given a point-clouds, $P = \{p_j\}_{j=1}^J \subset \mathbb{R}^n$, situated near a manifold \mathcal{M} in \mathbb{R}^n

1. The fill distance of the set P is defined as

$$h_0 = \max_{y \in \mathcal{M}} \min_{p_j \in P} \|y - p_j\|. \quad (2.3)$$

2. Given a point-set $Q = \{q_i\}_{i=1}^I \subset \mathbb{R}^n$, situated near a manifold \mathcal{M} in \mathbb{R}^n , such that its sizes and the size of the set P obey the constraint $I \leq J$, denote $\nu = \lfloor \frac{J}{I} \rfloor$. Then we say that the radius that guarantees approximately ν points from P in the support of each point q_i is $\hat{h}_0 = c_1 h_0$, with c_1 given by

$$\hat{h}_0 = c_1 h_0, \text{ with } c_1 = \operatorname{argmin}\{c : \#(\bar{B}_{c h_0}(q_i) \cap P) \geq \nu, \forall q_i \in Q\}. \quad (2.4)$$

Theoretical analysis of the MLOP method

For the sake of completeness, we summarize here several important properties of the MLOP algorithm. In [17] the authors proved that, under mild conditions, the non-convex optimization converges almost surely, with the number of iterations bounded, to a quasi-uniformly set Q^* which approximates the original manifold with a second-order approximation order. The following theorem summarizes all the theoretical results proven in [17]:

Theorem 2.3. *Let $\mathcal{M} \in \mathbb{R}^n$ be a C^2 manifold \mathcal{M} of unknown intrinsic dimension d . Suppose that the scattered data points $P = \{p_j\}_{j=1}^J$ were sampled from \mathcal{M} , h_1 and h_2 are set as in definition 2.2, and the h - ρ set condition is satisfied with respect to \mathcal{M} . Let the points $Q^{(0)} = \{q_i^{(0)}\}_{i=1}^I$ be sampled from P , and suppose Q^* is a local fixed-point solution of the gradient descent iterations. Then the following hold:*

- (1) *The gradient descent iterations (5.6) converge almost surely to a local minimizer Q^* .*

- (2) The proximity of the set Q^* to \mathcal{M} is $O(h^2)$ as $h \rightarrow 0$, where h is the support of the weight functions $w_{i,j}$.
- (3) The set Q^* is quasi-uniformly distributed with respect to \mathcal{M} .
- (4) The set Q^* is an ϵ -first-order stationary point, i.e. $\|\nabla f(x)\| \leq \epsilon$, that will be reached after $k = L \cdot (G(Q^{(0)}) - G(Q^*))\epsilon^{-2}$ iterations, for $\epsilon > 0$, step size $\gamma = \epsilon^{-1}$, and the cost function G is an L -smooth function and L is its Lipschitz constant.
- (5) The complexity of the algorithm is $O(nmJ + kI(nm\hat{I} + \hat{J}))$, where the number of iterations k is given in the item above, $m \ll n$ is the smaller dimension to which we reduce the dimension of the data, and \hat{I} and \hat{J} are the numbers of points in the support of the weight functions $\hat{w}_{i,i'}$, $w_{i,j}$ that belong to the Q -set and P -set, respectively. Thus, the approximation is linear in the ambient dimension n , and does not depend on the intrinsic dimension d .

Often, that data is incomplete, and contain holes. This scenario was addressed in [16], where missing information in the data is reconstructed.

2.2. Radial Basis Functions

Radial Basis Functions (RBF) constitute a very useful and convenient multivariate interpolation tool [15, 7]. Assume $\mathbb{X} = \{x_i\}_{i=1,\dots,K}$ is a finite set of distinct points in \mathbb{R}^d , which are traditionally called centres, because our basis functions will be radially symmetric about these points. Let us consider the function $f : \mathbb{R}^d \rightarrow \mathbb{R}^s$, its values these centers, $f(x_i)$. The goal is to approximate the value of the function in a new point $x \in \mathbb{R}^d$. In order to approximate this function in high-dimension, the approximation is carried out componentwise in the s components of f . The RBF estimates the value of f at a new point x by the formula

$$\tilde{f}(x) = \sum_{i=1}^K \mu_i \phi(\|x - x_i\|), \quad (2.5)$$

where $\phi : \mathbb{R}_+ \rightarrow \mathbb{R}$ is a radial basis function and μ_i is a vector of parameters chosen to maintain interpolation at the center points, i.e.,

$$\tilde{f}(x_j) = \sum_{i=1}^K \mu_i \phi(\|x_j - x_i\|) = f(x_j). \quad (2.6)$$

Several examples of radial basis functions appear in [8, 33]. In the numerical experiments, presented below, we used the following fast decaying Gaussian RBF:

$$\begin{aligned} \phi_1(r) &= \exp\{-(r/h)^2\}, \\ \phi_2(r) &= \exp\{-(r/h)^2\}(1 + r/h), \end{aligned}$$

In what follows, will discuss the order of approximation of the RBF algorithm, by stating a theorem proved in [34]. The theorem addresses a general form of the RBF,

where the polynomial term $\sum_{i=1}^I \alpha_i p_i(x)$ is added to equation 2.5. We narrow down our discussion to the case where $\alpha_i = 0$.

To state the theorem on the order of approximation of the RBF method we define below quasi-uniformly distributed point-sets.

Definition 2.4. Let $X = \{x_j\}_{j=1}^K \in \mathbb{X}$ be a set of points with fill-distance h and separation distance $\delta = \min_{1 \leq i \neq j \leq N} \|x_i - x_j\|/2$. Then we say that the point-sets $X \in \mathbb{X}$ are quasi-uniformly distributed if there exists a constant $\eta > 0$ independent of X such that

$$2\delta \leq h \leq \eta\delta. \quad (2.7)$$

The approximation theorem presented in [34] deals with a family \mathbb{X} of quasi-uniformly distributed point-sets in \mathbb{R}^d , stating the approximation order w.r.t. their fill-distance h .

Theorem 2.5. Let \mathbb{X} be a family of quasi-uniformly distributed scattered point-sets, and let $\tilde{f}(x)$, defined as in (2.5), be an interpolant to f on $X \in \mathbb{X}$ using the radial basis function ϕ . Also let $W_\infty^k(\Omega)$ be the Sobolev space, for $k \in \mathbb{N}$. Then under mild assumptions on ϕ , for every function $f \in W_\infty^k(\Omega)$, with $k \in \mathbb{N}$, the error of the RBF method is estimated as

$$\|f - \tilde{f}\|_{L_\infty(\Omega)} = o(h^k) \text{ as } h \rightarrow 0. \quad (2.8)$$

Remark 2.6. While the above approximation result holds for functions on \mathbb{R}^d , we are going to apply the RBF procedure on smooth d -dimensional manifolds. To transfer the approximation rate result to the approximation of functions on manifolds, we need some further assumptions on the smoothness of the manifold and on the RBF basis function. In the following proposition we present a partial discussion on the approximation rate of the RBF approximation on manifolds, under some restricted assumptions.

Proposition 2.7. RBF on manifolds. Let \mathcal{M} be a C^2 d -dimensional manifold in \mathbb{R}^n , and let $\rho(x, y)$ represent the geodesic distance between two points on \mathcal{M} . Consider $f : \mathcal{M} \rightarrow \mathbb{R}$, $f \in W_\infty^k$ on \mathcal{M} with respect to ρ , $k \leq 2$. Assume f is given on a set of points $Q \in \mathcal{M}$, quasi-uniformly distributed on \mathcal{M} , of fill-distance h , with respect to ρ . Consider the manifold RBF approximation at a point $z \in \mathcal{M}$ defined as

$$\tilde{f}(z) = \sum_{q_i \in Q} \mu_i \phi(\rho(z, q_i)), \quad (2.9)$$

where $\phi : \mathbb{R}_+ \rightarrow \mathbb{R}$ is a function of finite support and $\{\mu_i\}$ is the vector of parameters chosen to fulfill interpolation at the points Q ,

$$\sum_{q_i \in Q} \mu_i \phi(\rho(q_i, q_j)) = f(q_j), \quad q_j \in Q. \quad (2.10)$$

Then, as $h \rightarrow 0$,

$$|f(z) - \tilde{f}(z)| = o(h^k). \quad (2.11)$$

Proof. Consider the system of equations in (2.6) in case ϕ is of finite support h , or fast decaying as ϕ_1 and ϕ_2 above. Let us denote the data vector as $\bar{f} \equiv \{f(x_j)\}$, and the resulting coefficients vector as $\bar{\mu} \equiv \{\mu_i\}$. Then,

$$\bar{\mu} = B\bar{f},$$

where B is the inverse of the matrix $\{\phi(\|x_i - x_j\|)\}$. Using the same consideration as in [13], we have that $|B_{k,\ell}| \leq \exp^{-C\frac{\|x_k - x_\ell\|}{h}}$. It follows that essentially only data in an $O(h)$ neighborhood of $x \in \mathbb{R}^n$ affects the approximation $\tilde{f}(x)$. The same holds for the RBF approximation on manifolds, i.e., the approximation order at $z \in \mathbb{R}^n$ is affected by the data in an $O(h)$ neighborhood of z .

In order to transfer the approximation result in Theorem 2.5 to the manifold setting, we consider the projection Π of a small neighborhood $\mathcal{N} \subset \mathcal{M}$ of $z \in \mathbb{R}^n$ onto the tangent space to \mathcal{M} at that point. The projection induces the definition of a function \hat{f} on $\Omega = \Pi(\mathcal{N})$ simply by $\hat{f}(\Pi(q)) = f(q)$, $q \in \mathcal{N}$. Since \mathcal{M} is C^2 , it follows that $\hat{f} \in W_\infty^k$ on Ω . We also consider the projection of the data points, $\hat{q}_j = \Pi(q_j)$, $q_j \in \mathcal{N}$. The ordinary RBF approximation on Ω uses the values $\hat{f}(\hat{q}_j) = f(q_j)$ and the weights $\phi(\|\hat{q}_i - \hat{q}_j\|)$. For the RBF approximation on Ω we are equipped with the approximation result of Theorem 2.5. Since \mathcal{M} is C^2 , it follows that for data points in an $O(h)$ neighborhood of z

$$\phi(\|\hat{q}_i - \hat{q}_j\|) = \phi(\rho(q_i, q_j)) + O(h^2), \text{ as } h \rightarrow 0. \quad (2.12)$$

Let us compare the linear systems of equations related to the RBF approximation on Ω and the RBF approximation on the manifold. In view of the above discussion on the relevant influence domains for both approximations at the point z , and in view of (2.12), it follows that the significant equations for the approximation at z using the RBF on \mathcal{M} are $O(h^2)$ perturbations of the corresponding equations for the RBF approximation on Ω . This implies an $O(h^2)$ perturbation in the solution for the coefficients defining the approximation at z . This, in turn, implies an $O(h^2)$ perturbation in the approximation rate result (2.8). Since $k \leq 2$, the proof is completed. \blacksquare

Remark 2.8. Limitations of the RBF method. Theorem 2.5 implies that in the ideal scenario of noise-free, uniformly-sampled data, the RBF method results in a good approximation rate. Nevertheless, in real-life scenarios, this is not always the case. From Theorem 2.5 it follows that the approximation order of the RBF function is $o(h^k)$, where h is the fill-distance of the scattered data. However, non-uniform sampling as well as the presence of noise enlarges the fill-distance and reflects the approximation rate. In addition noise in the function values, naturally, affect the approximation rate as well. The approach presented in the next section suggests a methodology that solves these issues.

§3. Function Approximation via FMLOP Method

3.1. Algorithm Description

Let \mathcal{M} be a manifold in \mathbb{R}^n , of unknown intrinsic dimension, where $d \ll n$, and let $f : \mathcal{M} \rightarrow \mathbb{R}^s$ be a C^k smooth function. Assume $\tilde{P} = \{\tilde{p}_i\}$ is a noise-free set of points sampled non-uniformly on \mathcal{M} , and let $\{e_i\}$ be a set of random vectors in \mathbb{R}^d with zero mean and $\|e_i\| \leq \epsilon$. Define a noisy point-cloud $P = \{p_j\}_{j=1}^J \subset \mathbb{R}^n$ as $p_i = \tilde{p}_i + e_i$, be a disjoint set (i.e., no repetitive points exist), situated near the manifold. Given noisy function values $f(p_j)$ at locations $P = \{p_j\}_{j=1}^J$, the approximation solution presented in this paper consists of two steps. First, find a quasi-uniform, noise-free point set that approximates the manifold \mathcal{M} , and recover the function values at these points. Later, given a new point z near \mathcal{M} , approximate the value of the function f at it. The approach proposed herein leverages the advantages of the Manifold Locally Optimal Projection method to deal with issues raised in Remark 2.8. First, the available data (scattered data as well as the function values) is denoised via the MLOP method. Once the data cleansing is complete, it is used to estimate the function value at new points via the RBF method. It should be emphasized that data reconstruction and denoising is a pre-processing step that should be performed only once for each input scenario, and later we enjoy its fruits for estimating the function at any given new point. In what follows we use the following notations: P and Q are data sets in \mathbb{R}^n , while \hat{P} and \hat{Q} are sets in \mathbb{R}^{n+s} , where s is the dimension of the codomain of the function.

Step 1 - Data Denoising

Our approach for the first step is based on generalizing the MLOP method designed for manifold denoising to the case of function approximation settings. The key idea of the solution is embedding the approximation problem in a higher-dimensional space and carrying out the denoising task for the embedded data. Given input data, consisting of the points set $P = \{p_j\}_{j=1}^J \subset \mathbb{R}^n$ and the set of the corresponding function values at them $\{f(p_j)\}_{j=1}^J \subset \mathbb{R}^s$, we define a new point-set $\hat{P} = \{\hat{p}_j\}_{j=1}^J$ as the graph of the function f on P , as

$$\hat{p}_j = (p_j, f(p_j)). \quad (3.1)$$

From the definition it follows that the newly defined points \hat{P} are data points in \mathbb{R}^{n+s} . Noteworthy is the data standardization step which is an essential phase prior to defining the points \hat{p}_j , see Remark 3.2 for a complete discussion of this delicate point.

Lemma 3.1. *Assume \mathcal{M} is a smooth d -dimensional manifold in \mathbb{R}^n , where $d \ll n$ and let $f : \mathcal{M} \rightarrow \mathbb{R}^s$ be a C^k smooth function. Let $\widehat{\mathcal{M}} = \text{Graph } f = \{(x, f(x)) : x \in \mathcal{M}\} \subset \mathbb{R}^{n+s}$, then $\widehat{\mathcal{M}}$ is a smooth d -dimensional manifold in \mathbb{R}^{n+s} .*

Proof. From the definition of $\widehat{\mathcal{M}}$, it follows that it is the graph of a smooth function f defined on a smooth d -dimensional manifold. Therefore, there are only d free parameters that define $\widehat{\mathcal{M}}$, and thus it is itself a smooth d -dimensional manifold in \mathbb{R}^{n+s} . ■

In this context, the points set \widehat{P} represents noisy data sampled from a d -dimensional manifold embedded in the \mathbb{R}^{n+s} . In the new setting applying the MLOP method on the new data set \widehat{P} results in a clean, quasi-uniformly sample set $\widehat{Q} = \{\widehat{q}_i\}_{i=1}^I \subset \mathbb{R}^{n+s}$ which serve as a noise-free approximation of $\widehat{\mathcal{M}}$. From Theorem 2.3, the set \widehat{Q} has several properties that will be later used in the theoretical analysis. First, it follows that the set \widehat{Q} is a quasi-uniform approximation to $\widehat{\mathcal{M}}$, which is a cornerstone for the second RBF approximation step below (see the discussion in Remark 2.8). In addition, applying the MLOP in the high dimensional case denoises the entire data efficiently. Due to our construction the denoising is performed both the domain of the function and its co-domain.

Let us examine the end product of the MLOP reconstruction, \widehat{Q} , and its structure. The construction of the set \widehat{P} in (3.1) determines the structure of \widehat{Q} . Thus, each point \widehat{q}_i can be rewritten as

$$\widehat{q}_i = (q_i, \widetilde{f}(q_i)), \quad i = 1, \dots, I. \quad (3.2)$$

Let us define a points set $Q = \{q_i\}_{i=1}^I \subset \mathbb{R}^n$. This set is recovered as the set of the first n component of each point in the graph, while the last s components represent the denoised values of f at these points. From Theorem 2.3 it follows that it approximates the original manifold \mathcal{M} .

Step 2 - Approximation of the Function at a New Point

Given the denoised data set $(q_i, \widetilde{f}(q_i))$, we now address the problem of estimating the function at a new point z near \mathcal{M} by applying the RBF approximation taking into account the denoised data. Since the outcome of the first step is a set of points Q which is noise-free, and is also quasi-uniformly distributed on the manifold \mathcal{M} , applying the RBF using these points, is beneficial as opposed to non-uniformly sampled points (see Remark 2.8). This key idea paves the way towards estimating the value of the function at a new given point on \mathcal{M} , or near \mathcal{M} , with a satisfactory order of approximation.

In what follows, we approximate f using the RBF approximation (2.5), while setting the centers at the noise-free points Q , and relying on the noise-free function values $\widetilde{f}(Q)$. Thus, given a new point $z \in \mathbb{R}^n$, f is estimated by the formula

$$\widetilde{f}(z) = \sum_{i=1}^I \mu_i \phi(\|z - q_i\|), \quad (3.3)$$

where q_i are the points found by the MLOP in Step 1 as appearing in 3.2, and μ_i are found by solving the system of equations (2.6),

$$\sum_{i=1}^I \mu_i \phi(\|q_{i'} - q_i\|) = \widetilde{f}(q_{i'}), \quad q_{i'} \in Q. \quad (3.4)$$

The steps of the FMLOP algorithm 1 are summarized below:

Algorithm 1 Function Approximation on a Manifold in High Dimensions (FMLOP)

- 1: **Input:** $P = \{p_j\}_{j=1}^J \subset \mathbb{R}^n$, $\{f(p_j)\}$, $\{z_k\}_{k=1}^K$ ▷ where $\{z_k\}_{k=1}^K$ is a set of new points for function approximation
 - 2: **Output:** $Q = \{q_i\}_{i=1}^I \subset \mathbb{R}^n$, $\tilde{f}(Q)$, $\{\tilde{f}(z_k)\}_{k=1}^K$
 - 3: Normalize the P points and the $f(P)$ values to a common scale (see Remark 3.2) .
 - 4: Embed the data to \mathbb{R}^{n+s} as the graph of the function $\hat{P} = (P, f(P)) \subset \mathbb{R}^{n+s}$.
 - 5: Denoise the input data by running MLOP with \hat{P} that result with $\hat{Q} = (Q, \tilde{f}(Q))$.
 - 6: **for** each $z_k \in \{z_k\}_{k=1}^K$ **do**
 - 7: Approximate $\tilde{f}(z_k)$ via RBF (3.3), with centers at Q and the matching $\tilde{f}(Q)$ values.
 - 8: **end for**
-

Remark 3.2. Data standardization. Distance calculation is the cornerstone of the MLOP algorithm. It is known that the vector norm is tightly dependent on the scale of its entries. Therefore, when dealing with high-dimensional data, one needs to make sure that the data in use is homogeneous (i.e. that the entries of each sample are from the same numerical range). This is a basic requirement not only in our case but also in many manifold learning and machine learning tasks when data acquired from different domains are combined. There exist several ways to perform data normalization, starting from standardization (by normalizing the features utilizing the mean and variance), such as min-max normalization, or unit vector normalization.

In our case, upon embedding the data in a higher-dimensional space as the graph of the function, one needs to address the data normalization problem. Since naturally the points p_j and their function values $f(p_j)$ are not necessarily on the same scale, performing a normalization is an essential step to avoid the dominance of one over the other. In the numerical examples, we choose to normalize the values of f to match the scale of the P coordinates, by scaling the function values with respect to the maximum absolute value of the p_j coordinates.

3.2. Theoretical Analysis of the FMLOP Method

In this section, we discuss some of the theoretical aspects of the proposed approximation method. Our analysis is incremental and relies on the theory of the MLOP method as discussed in Theorem 2.3, as well as the RBF method. Naturally, some of the properties of the FMLOP are inherited from the MLOP method, we summarize them in the following theorem:

Theorem 3.3. *Let $P = \{p_j\}_{j=1}^J$ be a set of points sampled from a d -dimensional C^2 manifold \mathcal{M} without noise that satisfy the h - ρ condition. Let $f : \mathcal{M} \rightarrow \mathbb{R}^s$ be a smooth multivariate function given at points of P . Let \hat{P} be the graph of the function defined in 3.1 sampled from a manifold $\widehat{\mathcal{M}}$ in \mathbb{R}^{n+s} . Assume \hat{h}_1 and \hat{h}_2 are tightly related to the fill-distance of the dataset \hat{P} , as prescribed in Definition 2.2, and also the h - ρ set condition is satisfied with respect to $\widehat{\mathcal{M}}$. Then the MLOP converges almost surely to a local minimizer, the reconstructed set \hat{Q} is evenly distributed with respect to the manifold*

$\widehat{\mathcal{M}}$, and the MLOP reconstruction is reached after a bounded number of gradient descent iterations.

The proof of the theorem follows immediately from the fact that from Lemma 3.1, the \widehat{P} set was sampled from a manifold in \mathbb{R}^{n+s} . Thus, all the conditions in Theorem 2.3 are satisfied, and the proof follows for the \widehat{P} and \widehat{Q} data.

Next, we build on top of the MLOP results concerning the order of approximation and the complexity of the method to analyse the FMLOP properties. It should be noted that although the theoretical analysis deals with noise-free samples, in the numerical examples section we illustrate the application of the algorithm to noisy data.

Theorem 3.4 (Order of Approximation). *Let $P = \{p_j\}_{j=1}^J$ be a set of points sampled from a d -dimensional C^2 manifold \mathcal{M} without noise that satisfy the h - ρ condition. Let $f : \mathcal{M} \rightarrow \mathbb{R}^s$ be a smooth multivariate function given at points of P . Suppose that $f \in W_\infty^k(\Omega)$, $k < 2$, and fulfills all the conditions of Theorem 2.5. Let $\widehat{P} = \{(p_j, f(p_j))\} \subset \mathbb{R}^{n+s}$ be the corresponding graph of the function f on P . Assume \widehat{h}_1 and \widehat{h}_2 are tightly related to the fill-distance of the dataset \widehat{P} , as prescribed in Definition 2.2. Let also $\widehat{Q} = \{(q_i, \widetilde{f}(q_i))\}$ be the pair of $Q = \{q_i\}_{i=1}^I \subset \mathbb{R}^n$ the points set that reconstruct \mathcal{M} that result from the FMLOP method, accompanied with the $\widehat{f}(Q)$. Then the following holds*

1. For a fixed ρ and δ , the set Q approximates \mathcal{M} with order $O(\widehat{h}^2)$, where $\widehat{h} = \max\{\widehat{h}_1, \widehat{h}_2\}$. In addition, the order of approximation of the function f on the set Q is also $O(\widehat{h}^2)$.
2. The error of approximating the function value at a new point using the RBF approximation with centers at Q , and the values of $\widehat{f}(Q)$, is bounded by $C_1 \widehat{h}^2 + C_2 h_2^k$, where \widehat{h}_2 is found with respect to the set \widehat{Q} , h_2 is estimated taking in to account the set Q , and C_1 and C_2 are constants.

Proof. Given the input data, consisting of the points $P = \{p_j\}_{j=1}^J$ and the function values $\{f(p_j)\}_{j=1}^J$, we follow the FMLOP procedure. Let us define a new point-set $\widehat{P} = \{\widehat{p}_j\}_{j=1}^J$, which lie on the graph of the function f . By Lemma 3.1 it follows that these points are sampled from the manifold $\widehat{\mathcal{M}}$ in \mathbb{R}^{n+s} . In this setting, applying the MLOP algorithm denoise the d -dimensional manifold embedded in $(n+s)$ -dimension. Let \widehat{h} be the representative distance introduced in Definition 2.2 for the data set $\widehat{P} = \{(p_j, f(p_j))\} \in \mathbb{R}^{n+s}$. From Theorem 2.3 it follows that \widehat{Q} approximates $\widehat{\mathcal{M}}$ with the order $O(\widehat{h}^2)$, and also that the reconstruction is a quasi-uniformly distributed set \widehat{Q} . Recall that \widehat{Q} is in fact a combination of a noise-free set Q , which reconstructs the manifold \mathcal{M} , and an estimate of the clean values of the function at these points in Q , $\widetilde{f}(Q)$. Therefore, the order of approximation of Q to \mathcal{M} , and of the estimated values \widehat{f} to f is $O(\widehat{h}^2)$. This proves the first part of the Theorem.

Next, let h_2 be the the support size of $\widehat{w}_{i,i'}$ as in Definition 2.2 with respect to the point-set Q . From Proposition 2.7 it follows that the error of approximation of the function value at a new point is $o(h_2^k)$. Therefore, the overall order of approximation

for a new point is a combination of the two orders, namely $\leq C_1 \widehat{h}^2 + C_2 h_2^k$, with C_1 and C_2 are constants. \blacksquare

Remark 3.5. The fill-distance \widehat{h} of the set $\widehat{P} \subset \mathbb{R}^{n+s}$ defined as in 2.2 is related to the construction of that set in 3.1. On the one hand, it depends on the local density of the dataset P , and on the other hand, it also incorporates the information on function variation.

We now turn to discuss the complexity of the approximation of functions. The complexity consists of a pre-processing MLOP stage performed only once, and the RBF approximation of the function at new points.

Theorem 3.6 (Complexity of the Approximation of Functions). *Let $P = \{p_j\}_{j=1}^J$ be a set of points sampled near a d -dimensional manifold $\mathcal{M} \subset \mathbb{R}^n$ and let $f : \mathcal{M} \rightarrow \mathbb{R}^s$ be a multivariate function given at the points of P . Let us apply the MLOP iterations k times, which will result in the points $Q = \{q_i\}_{i=1}^I$. Let m be $m \ll n$, the dimension of the domain where the linear sketching reduces the data. Then the complexity of function approximation via the MLOP is $O((n+s)mJ + kI((n+s)m\widehat{I} + \widehat{J}) + nmI + I \log_2^2 s)$, where \widehat{I} and \widehat{J} are the numbers of active points from the Q and P -sets, respectively. In addition, the complexity of evaluating the function at a new point is $O(nm + I)$.*

Proof. The estimate of the complexity of the algorithm can be separated into two steps: a) pre-processing and b) evaluating the function at a new point. The pre-processing step consists of applying the MLOP algorithm in \mathbb{R}^{n+s} , as well as finding the RBF coefficients μ_i by solving the Least-Squares problem in 2.6. By Theorem 2.3, the complexity of applying the MLOP in the higher dimension $n+s$ is $O((n+s)mJ + kI((n+s)m\widehat{I} + \widehat{J}))$. The output of this stage is a point set Q of size I , and the corresponding set of values $\widehat{f}(Q)$. From this point on, all the approximation operations are performed on Q , and if $I \ll J$ then we can increase sufficiently the efficiency. The next part of the pre-processing step is to evaluate the radial basis function ϕ for each $q_i \in Q$, which costs $O(nmI)$, and then to find the μ_i by solving the Least-Squares problem in 2.6, which takes $O(I \log_2^2 s)$ (as shown in [23]). As a result, the complexity of the pre-processing step is $O((n+s)mJ + kI((n+s)m\widehat{I} + \widehat{J}) + nmI + I \log_2^2 s)$.

Finally, using the μ_i already found, we evaluate the function at a new point in time $O(nm + I)$. It should be stressed that although the pre-processing steps are expensive, they are executed once before the function is approximated at new points set. Thus, if the number of new points for which the approximation needs to be found is large, then the pre-processing steps have less effect on the runtime. \blacksquare

§4. Numerical Examples

In what follows we present several numerical experiments to demonstrate the effectiveness and advantages of our methodology. Our study includes various types of functions evaluated on manifolds of different intrinsic dimensions, among them are:

- a. Smooth/non-smooth function on a one-dimensional manifold of orthogonal matrices embedded in \mathbb{R}^{60} .

- b. A sinusoidal function on a two and six-dimensional cylinder in \mathbb{R}^{60} .
- c. A function with a codomain in \mathbb{R}^2 on a two-dimensional cone in \mathbb{R}^{60} .
- d. Robustness test with various noise levels of a function on a Swiss Roll embedded in \mathbb{R}^{60} .

Our examples illustrate the strengths of the proposed approximation approach. On the one hand, denoising the data domain as well as the function codomain plays an important role in the approximation of functions. On the other hand, sampling the manifold quasi-uniformly improves significantly the approximation of functions using classical approximation methods on new data. It should be emphasized that in all the numerical examples the noise was added both to the data and the observed function values. The accuracy of the approximation is measured as the relative maximum error of the L_1 norm of the difference between the value of \tilde{f} at the new point and the value of f at the closest point in the reference dataset, as well as the root-mean-square error and the standard deviation. In addition, for evaluating the efficiency of approximating function at new points, we randomly selected 100 points.

4.1. Smooth/ Non-Smooth Functions on One-Dimensional Orthogonal Matrices

We start with two examples of functions, one smooth and the other non-smooth, both calculated on a one-dimensional manifold embedded in a high-dimensional space. Although in principle the approximation requires a smooth function, it can still be applied to a non-smooth function, provided that we end up with a smoothed result. In this experiment, we sampled 500 equally distributed points from the manifold $O(2)$ of orthogonal matrices

$$\hat{p}_\theta = [\cos(\theta), -\sin(\theta), \sin(\theta), \cos(\theta), 0, \dots, 0]$$

for $\theta \in [-\pi, \pi]$. Next, we projected the data to a non-trivial domain using randomly generated orthogonal matrix $A \in \mathbb{R}^{60 \times 60}$, resulting in a new point-set via the non-trivial vector embedding $P = A\hat{P}$, and added uniformly distributed (between -0.1 and 0.1) additive noise. Next, the set Q was formed from 55 points randomly sampled. Figure 1 (A) illustrates the first two coordinates of the points, after multiplication by the matrix A^{-1} . The noisy sample points are shown in blue, while the initial Q points are in red.

We studied two functions on this manifold. The first one was a smooth function given in the following parameterization

$$f(x) = \frac{1}{4}(1 + \sin(10\theta)),$$

while the other function was the non-smooth function given by

$$f(x) = \frac{1}{6}(1 + \arccos(\cos(10\theta))).$$

In both of the cases, θ corresponds to the value used in the definition of \hat{p}_θ above. The two functions were evaluated at the P -points and uniform noise $U(-0.1, 0.1)$ was added, see Figure 1 (B) and (C) respectively. Later the MLOP algorithm was applied, which reconstructed and denoised the manifold (see Figure 1 (D)), as well as denoised function values; see Figure 1 (E) and (F). The comparison between the approximation and the original as presented in Figure 1 speaks for itself. This is also reflected in the errors listed in Table 1, which shows that the maximal approximation error decreased from 0.31, and 0.13 for the smooth function and from 0.2 to 0.1 for the non-smooth one.

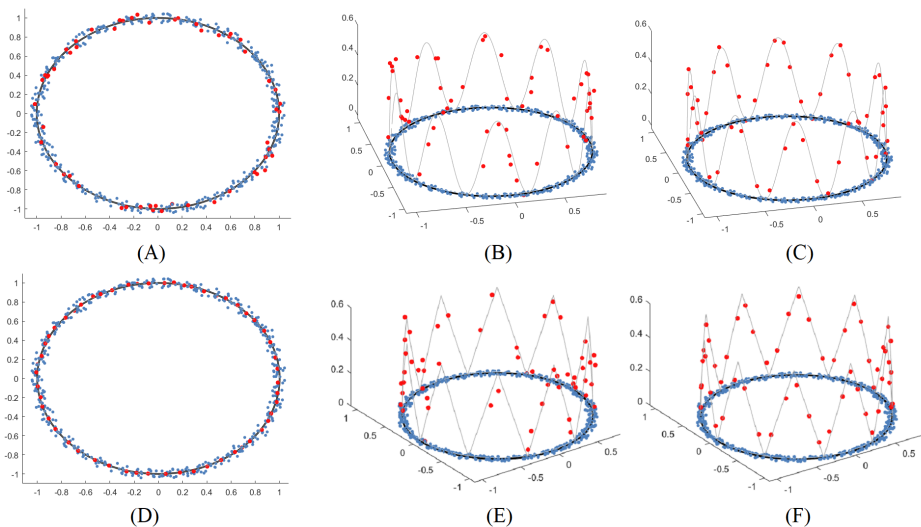


Figure 1: Approximating sinusoidal and zigzag functions on an orthogonal matrices manifold embedded in \mathbb{R}^{60} . Displayed using the first two coordinates of the point-set (after multiplication by A^{-1}) (A); Scattered data with uniformly distributed noise in blue, and the initial point-set in red; Noisy function values in red and the noise-free reference data in gray. The smooth case presented in (B); and non-smooth in (C); (D) The denoised manifold resulting after the MLOP algorithm after 150 iterations in red; Noise-free functions values of the smooth (E); and non-smooth function (F).

Table 1: Error estimation of function values $f(Q^{(k)})$ on the $O(2)$ manifold embedded into 60-dimensional space.

	$f(x) = \frac{1}{4}(1 + \sin 10x)$		$f(x) = \frac{1}{6}(1 + \arccos(\cos 10x))$	
	Max relative error	RMSE \pm var	Max relative error	RMSE \pm var
$k = 0$	0.31	0.06 ± 0.0011	0.2	0.04 ± 0.0007
$k = 150$	0.13	0.03 ± 0.0003	0.1	0.03 ± 0.0002

Next, we examined the effectiveness of Step 2 of the FMLOP algorithm, i.e. estimating the function values at new points. For this purpose, 100 new points were randomly chosen on the manifold, and the function was estimated at them using the RBF approximation while using the Q points. The results are summarized in Table 2. For instance, for the smooth case, we see that the maximal error was reduced from 0.66 to 0.12 for the smooth case, while for the non-smooth it was reduced from 0.77 to 0.15. This is also strengthened by the RNSE and variance calculation. This shows the robustness of the approximation process with respect to the clean data.

Table 2: Error estimation of the RBF approximation at 100 new data points using the denoised centers

	$f(x) = \frac{1}{4}(1 + \sin 10x)$		$f(x) = \frac{1}{6}(1 + \arccos(\cos 10x))$	
	Max relative error	RMSE \pm var	Max relative error	RMSE \pm var
RBF, ϕ_1 , centers at $Q^{(0)}$, noisy f	0.66	0.14 \pm 0.0079	0.77	0.16 \pm 0.010
RBF, ϕ_1 , centers at $Q^{(150)}$, cleaned \tilde{f}	0.12	0.03 \pm 0.0002	0.15	0.03 \pm 0.0004
RBF, ϕ_2 , centers at $Q^{(150)}$, cleaned \tilde{f}	0.12	0.03 \pm 0.0002	0.16	0.03 \pm 0.0003

4.2. Two and Six-Dimensional Cylindrical Structure

In this subsection, we study the two and six-dimensional cylinder embedded to \mathbb{R}^{60} . We start with the two-dimensional cylindrical structure, sampled using the parameterization

$$p = tv_1 + \frac{R}{\sqrt{2}}(\cos(u)v_2 + \sin(u)v_3),$$

where $v_1 = [1, 1, 1, 1, 1, \dots, 1]$, $v_2 = [0, 1, -1, 0, 0, \dots, 0]$, $v_3 = [1, 0, 0, -1, 0, \dots, 0]$ ($v_1, v_2, v_3 \in \mathbb{R}^{60}$), $t \in [0, 2]$ and $u \in [0.1\pi, 1.5\pi]$. Here we sampled 800 equally distributed (in parameter space) points and added uniformly distributed noise $U(-0.1, 0.1)$. The function used is $f(t, u) = 1.3(1 + \sin(0.5u + 1.5t))$, and additive noise was added $U(-0.1, 0.1)$. The set Q consisted of 150 points randomly sampled from P , see Figure 2 (A) for the manifold and (B) for the function values illustrated by the color.

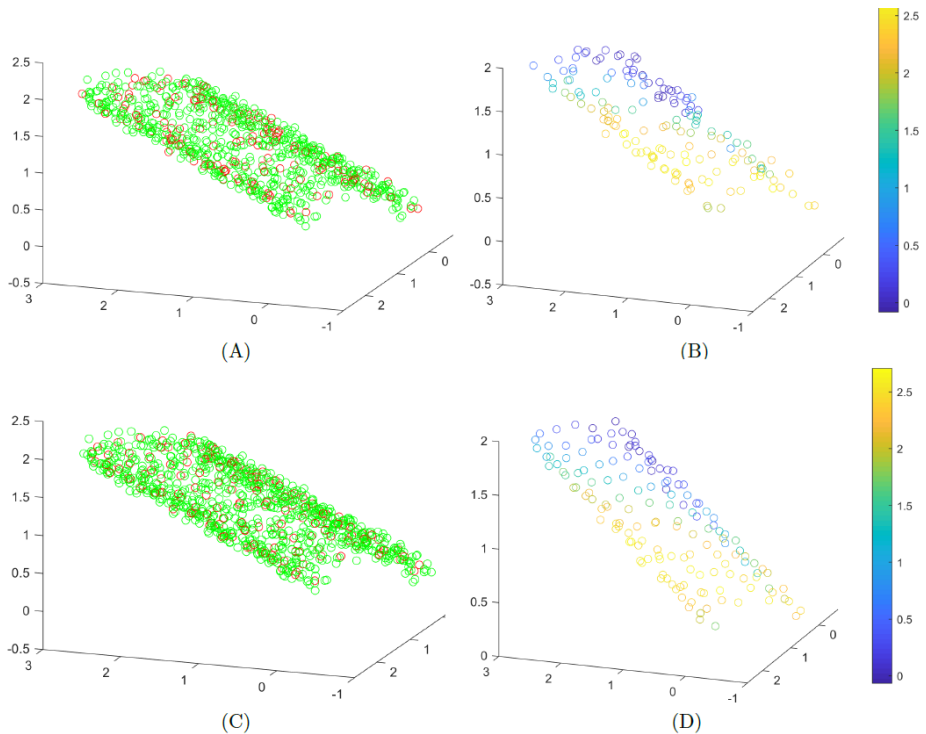


Figure 2: Approximating sinusoidal function on a two-dimensional cylindrical structure embedded in a 60-dimensional space. The first three coordinates of the point-set are shown. (A) Scattered data with uniformly distributed noise $U(-0.1; 0.1)$ (green), and the initial point-set $Q^{(0)}$ (red). (B) The initial values of the function at the original $Q^{(0)}$ -points with noise $U(-0.1, 0.1)$. (C) The resulting point-set of the MLOP algorithm after 300 iterations (red), overlays the noisy samples (green). (D) function approximation at the data points $Q^{(300)}$.

The results of applying the first step of the FMLOP are presented in Figure 2 bottom. In plot (C) the denoised Q -points data is shown. We see that the cylinder was reconstructed uniformly, and that the data, as well as the function values, were denoised. In plot (D) the cleaned function values at the new Q -points are presented, it's apparent that the color transition is smooth, indicating denoised function values. This is also apparent in the error analysis in Table 3 left column, where the maximal relative error decrease from 0.11 to 0.06.

Next, we approximated the values of f at 100 new points, randomly selected from the reference data. The evaluation error results are summarized in Table 4. It should be noted that since we compare the maximum error to clean data, the relative error can exceed 1. The improvement in the approximation is apparent with an error of 1.37 reduced to 0.11 achieved using ϕ_2 .

Table 3: Error estimation of function values $f(Q^{(k)})$ on the two-dimensional and six-dimensional cylindrical manifold embedded into 60-dimensional space.

	2D cylinder in \mathbb{R}^{60}		6D cylinder in \mathbb{R}^{60}	
	Max error	relative RMSE \pm var	Max error	relative RMSE \pm var
$k = 0$	0.11	0.09 ± 0.0029	0.066	0.08 ± 0.0018
$k = 300$	0.06	0.05 ± 0.0012	0.054	0.06 ± 0.0012

Six-Dimensional Cylindrical Structure

Next, we tested our method on higher-dimensional manifolds by utilizing an n -sphere to generate an $(n + 1)$ -dimensional cylinder. Using a five-dimensional sphere we build a six-dimensional manifold, parameterized as

$$x_1 = R \cos(u_1), \quad x_2 = R \sin(u_1) \cos(u_2), \quad \dots, \quad x_6 = R \sin(u_1) \sin(u_2) \cdots \sin(u_5) \sin(u_6).$$

We then embedded the sampled data in a 60-dimensional space by calculating

$$p = tv_0 + R^2[x_1, x_2, x_3, x_4, x_5, x_6, 0, \dots, 0], \quad (4.1)$$

where $R = 1.5$, $t \in [0, 2]$, $u_i \in [0.1\pi, 0.6\pi]$, $v_0 \in \mathbb{R}^{60}$ is a vector with 1's in positions $1, \dots, d+1$ and 0 in the remaining positions. In this experiment we sampled 1200 equally distributed points on the six-dimensional cylindrical with additive noise $U(-0.2; 0.2)$, and 460 points were used for the reconstruction. The function that was approximated on this manifold was

$$f(u_1, \dots, u_6) = \sum_{i=1}^6 u_i.$$

Avoiding visualization of a six-dimensional manifold, in Figure 3 we plot the cross-section of the cylindrical structure in three dimensions in (A), as well as the initial function values at these points in (B).

We then applied the MLOP algorithm and reconstructed the manifold (as can be seen in (C)) and the function values at these points (D). Despite the high dimension, the manifold was reconstructed uniformly and the result of function denoising is impressive. The maximum error rate reduced from 0.066 to 0.054, as can be seen in Table 3.

In addition, we approximated the values of the function at 100 new points, randomly selected from the reference data. The maximum relative L_1 error and the RMSE accompanied with the variance are summarized in Table 4. One can see that the error reduced from 0.42 to 0.08 while using the ϕ_2 .

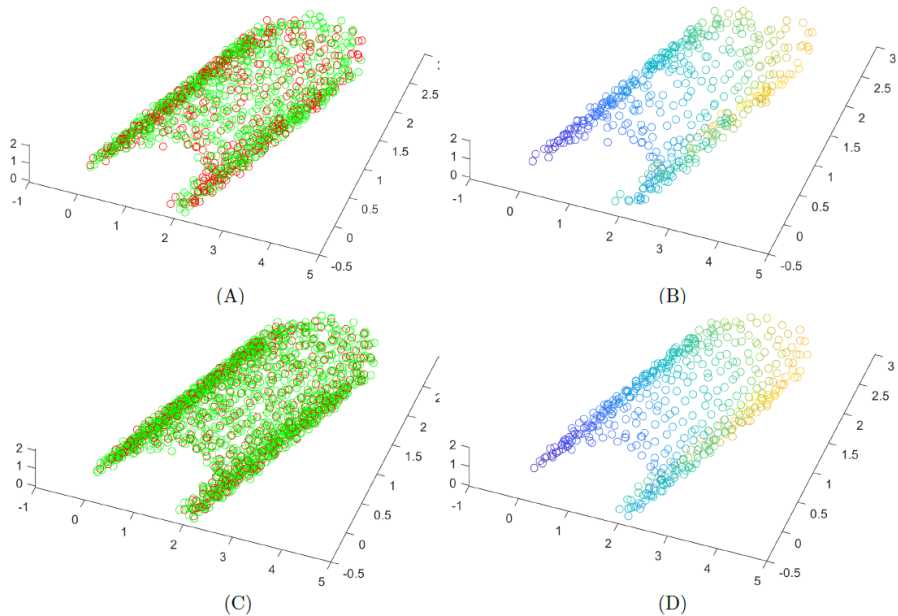


Figure 3: Approximating the sum function on a six-dimensional cylindrical structure embedded in a 60-dimensional space. The first three coordinates of the point-set are shown. (A) Scattered data with uniformly distributed noise $U(-0.1; 0.1)$ (green), and the initial point-set $Q^{(0)}$ (red). (B) The initial values of the function at the original $Q^{(0)}$ -points with noise $U(-0.1, 0.1)$. (C) The resulting point-set of the MLOP algorithm after 300 iterations (red), overlays the noisy samples (green). (D) function approximation at the data points $Q^{(300)}$.

Table 4: Error estimation of the RBF approximation at 100 new data points using the denoised centers

	2D cylinder in \mathbb{R}^{60}		6D cylinder in \mathbb{R}^{60}	
	Max relative error	RMSE \pm var	Max relative error	RMSE \pm var
RBF with ϕ_1 , centers at $Q^{(0)}$, noisy f	1.37	0.3 ± 0.04	0.42	0.4 ± 0.063
RBF with ϕ_1 , centers at $Q^{(300)}$, cleaned \tilde{f}	0.38	0.13 ± 0.007	0.26	0.26 ± 0.026
RBF with ϕ_2 , centers at $Q^{(300)}$, cleaned \tilde{f}	0.11	0.05 ± 0.0009	0.08	0.07 ± 0.0018

4.3. A 3D Function on a Three-Dimensional Cone in \mathbb{R}^{60}

Next, we demonstrate the strength of the FMLOP to cope with a geometric structure of different dimensions at different locations, as well as a function with non trivial codomain. Here we combined a cone structure, which is a 3-dimensional manifold, with a line segment. This geometry was embedded into a 60-dimensional linear space. The cone's parameterization used was

$$p = tv_1 + \frac{e^{-R^2}}{\sqrt{2}}(\cos(u)v_2 + \sin(u)v_3),$$

where $v_1 = [1, 1, 1, 1, 0, \dots, 0]$, $v_2 = [0, 1, -1, 0, 0, \dots, 0]$, $v_3 = [1, 0, 0, -1, 0, \dots, 0]$, $(v_1, v_2, v_3) \in \mathbb{R}^{60}$, $t \in [0, 2]$, $R \in [0, 2.5]$, and $u \in [0.1\pi, 1.5\pi]$. The function that was approximated on this manifold was the MATLAB peak function that uses the same u , and t values (normalised between -3 and 3)

$$f(t, u) = 3(1-t)^2 e^{-(t^2)-(u+1)^2} - 10(t/5 - t^3 - u^5) e^{-t^2-u^2} - 1/3 e^{-(t+1)^2-u^2}.$$

We sampled 880 points from this structure and added uniformly distributed noise of magnitude 0.2. The initial set $Q^{(0)}$ of size 60 was selected. In Figure 4 (A) the original cone structure is illustrated, along with the initial function values with noise in the magnitude of 0.5 (4 (B)). We performed 700 iterations of the MLOP were performed to reconstruct and denoise the geometrical structure, as well as denoise the function value. The results are plotted in Figure 4 (C) and (D) accordingly. One can see the noisy data in (C) in the upper part of the peak, that was denoised as appearing in (D). We see that the method deals with functions with a high codomain as well as with high levels of noise in a robust manner. Our numerical results show that the maximum error reduced from 1.42 to 0.14, that speaks for itself.

4.4. Robustness to Noise

In the following example, we examine the effect of the noise level in the target domain on the quality of the approximation. In this experiment we sampled a function over a Swiss Roll using the parameterization

$$p = \frac{1}{10}[x, y, z, 0, \dots, 0],$$

where $x = t \sin(t)$, y is a random number in the range $[-6, 6]$, and $z = t \cos(t)$, with $t = 8k/n + 2$ and $k \in \mathbb{N}$. The function we approximated was $f(p) = t$. We created a Swiss Roll with 800 data points and randomly sampled 200 points as the initial Q -set. We added noise with various magnitudes (0.1, 0.2, 0.5, and 0.7) to the P -points as well as to the values of f at the P -points. For example, Figure 5 left shows a case of approximation with uniformly distributed noise $U(-0.2, 0.2)$, while the right plot presents the denoised version. We see that the data was cleaned both in the domain and in the codomain of the function.

A summary of the effect of the noise level on approximation error is given in Figure 6. The plot consists of two graphes, one in orange that contain the errors for the noisy

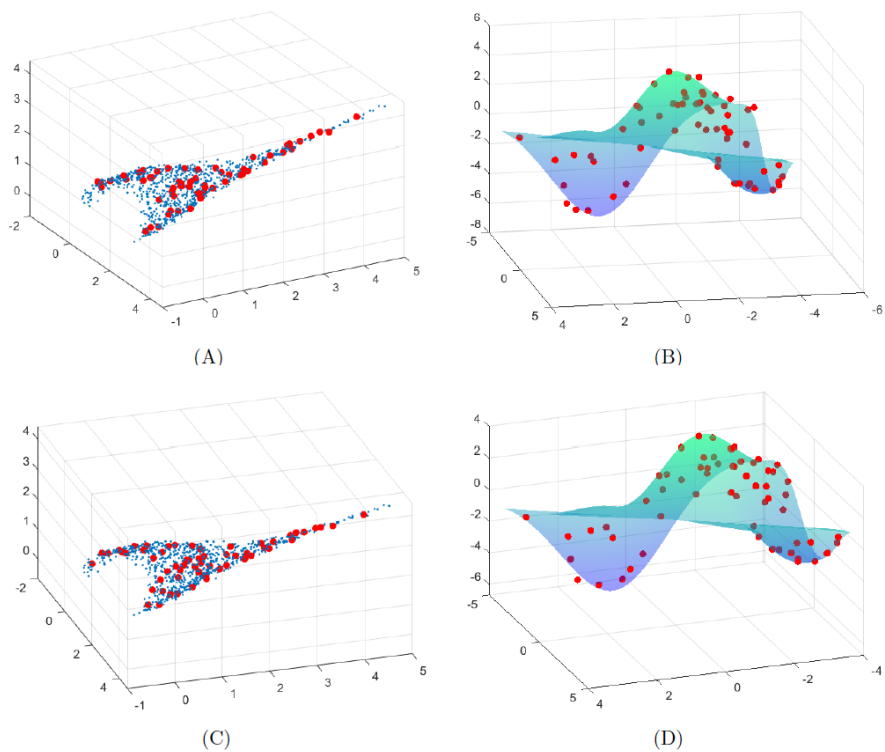


Figure 4: Approximating the peak function on a cone structure embedded to \mathbb{R}^{60} . The first three coordinates of the point-set are shown. (A) Scattered data with uniformly distributed noise $U(-0.1; 0.1)$ (blue), and the initial point-set $Q^{(0)}$ (red). (B) The initial values of the function with noise $U(-0.5, 0.5)$ on top of the reference clean data. (C) The resulting point-set of the MLOP algorithm (red), overlays the noisy samples (blue). (D) function approximation at the denoised data points.

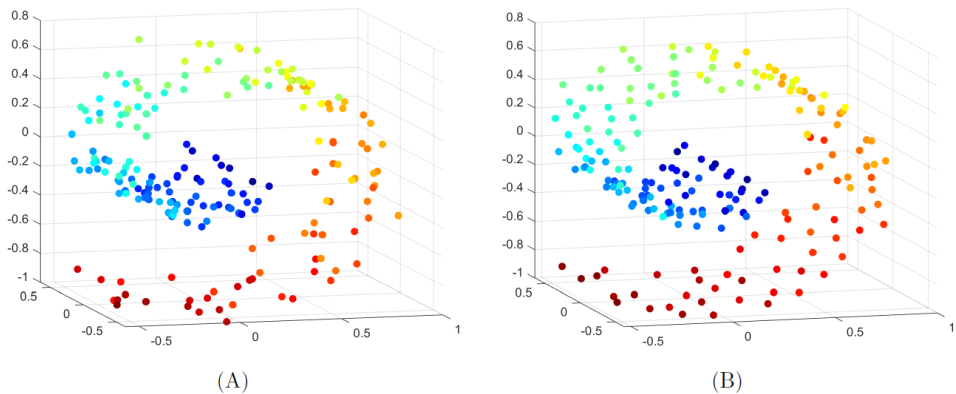


Figure 5: The first three coordinates of the Swiss Roll embedded in \mathbb{R}^{60} are presented. Left: The initial function values (represented with a color) at the original $Q^{(0)}$ points with noise $U(-0.2, 0.2)$. Right: MLOP function approximation at the data points $Q^{(300)}$ cleaned via the MLOP.

data ($f(Q^{(0)})$) and the other for the denoised function values ($f(Q^{(300)})$). As can be seen although the approximation error increases on the noisy data (from 0.03 to 0.22), the approximation error on clean and quasi-uniformly distributed data error grows slower (from 0.02 to 0.18). We also see that at high levels of noise (e.g., 0.7) the denoising operation as well as quasi-uniform sampling has more effect of the error.

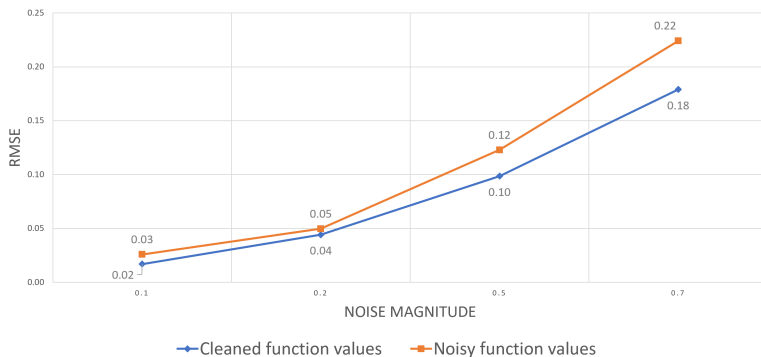


Figure 6: Effect of noise level on the accuracy of function approximation for a Swiss Roll embedded in a 60-dimensional space. The RMSE error evaluated on the original noisy data is shown in orange, while the RMSE error on the cleaned data is presented in blue.

§5. Appendix

The MLOP cost function is of the form

$$G(Q) = E_1(P, Q) + \Lambda E_2(Q) = \sum_{q_i \in Q} \sum_{p_j \in P} \|q_i - p_j\|_{H_\epsilon} w_{i,j} + \sum_{q_i \in Q} \lambda_i \sum_{q_{i'} \in Q \setminus \{q_i\}} \eta(\|q_i - q_{i'}\|) \widehat{w}_{i,i'}, \quad (5.1)$$

where the weights $w_{i,j}$ are rapidly decreasing smooth functions. The MLOP implementation uses $w_{i,j} = \exp\{-\|q_i - p_j\|^2/h_1^2\}$ and $\widehat{w}_{i,i'} = \exp\{-\|q_i - q_{i'}\|^2/h_2^2\}$. The “norm” $\|\cdot\|_{H_\epsilon}$ is introduced in [22] as $\|v\|_{H_\epsilon} = \sqrt{v^2 + \epsilon}$, where $\epsilon > 0$ is a fixed parameter (in our case we take $\epsilon = 0.1$). In addition, h_1 and h_2 are the support size parameters of $w_{i,j}$ and $\widehat{w}_{i,i'}$ which guarantee a sufficient amount of P or Q points for the reconstruction (for more details, see the subsection “Optimal Neighborhood Selection” in [17]). Also, $\eta(r)$ is a decreasing function such that $\eta(0) = \infty$; in our case we take $\eta(r) = \frac{1}{3r^3}$. Finally, $\{\lambda_i\}_{i=1}^I$ are constant balancing parameters.

We solve this problem via gradient descent iterations

$$q_{i'}^{(k+1)} = q_{i'}^{(k)} - \gamma_k \nabla G(q_{i'}^{(k)}), \quad i' = 1, \dots, I, \quad (5.2)$$

Where the gradient of G is given in the form of

$$\nabla G(q_{i'}^{(k)}) = \sum_{j=1}^J (q_{i'}^{(k)} - p_j) \alpha_j^{i'} - \lambda_{i'} \sum_{\substack{i=1 \\ i \neq i'}}^I (q_{i'}^{(k)} - q_i^{(k)}) \beta_i^{i'}, \quad (5.3)$$

with the coefficients $\alpha_j^{i'}$ and $\beta_i^{i'}$ given by the formulas where $\alpha_j^{i'}$ and $\beta_i^{i'}$

$$\alpha_j^{i'} = \frac{w_{i,j}}{\|q_i - p_j\|_{H_\epsilon}} \left(1 - \frac{2}{h_1^2} \|q_i - p_j\|_{H_\epsilon}^2 \right) \quad (5.4)$$

and

$$\beta_i^{i'} = \frac{\widehat{w}_{i,i'}}{\|q_i - q_{i'}\|} \left(\left| \frac{\partial \eta(\|q_i - q_{i'}\|)}{\partial r} \right| + \frac{2\eta(\|q_i - q_{i'}\|)}{h_2^2} \|q_i - q_{i'}\| \right), \quad (5.5)$$

for $i = 1, \dots, I$, $i \neq i'$. In order to balance the two terms in $\nabla G(q_{i'}^{(k)})$, the factors $\lambda_{i'}$ are initialized in the first iteration as

$$\lambda_{i'} = - \frac{\left\| \sum_{j=1}^J (q_{i'}^{(k)} - p_j) \alpha_j^{i'} \right\|}{\left\| \sum_{i=1}^I (q_{i'}^{(k)} - q_i^{(k)}) \beta_i^{i'} \right\|}. \quad (5.6)$$

Balancing the contribution of the two terms is important in order to maintain equal influence of the attraction and repulsion forces in $G(Q)$. The step size in the direction of the gradient γ_k is calculated as indicated in [3]:

$$\gamma_k = \frac{\langle \Delta q_{i'}^{(k)}, \Delta G_{i'}^{(k)} \rangle}{\langle \Delta G_{i'}^{(k)}, \Delta G_{i'}^{(k)} \rangle}, \quad (5.7)$$

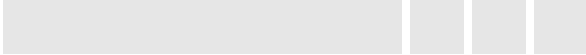
where $\Delta q_{i'}^{(k)} = q_{i'}^{(k)} - q_{i'}^{(k-1)}$ and $\Delta G_{i'}^{(k)} = \nabla G_{i'}^{(k)} - \nabla G_{i'}^{(k-1)}$.

References

- [1] Amir, A., Levin, D.: High order approximation to non-smooth multivariate functions. *Computer Aided Geometric Design* **63**, 31–65 (2018)
- [2] Andras, P.: High-dimensional function approximation with neural networks for large volumes of data. *IEEE Transactions on Neural Networks and Learning Systems* **29**(2), 500–508 (2017)
- [3] Barzilai, J., Borwein, J.M.: Two-point step size gradient methods. *IMA Journal of Numerical Analysis* **8**(1), 141–148 (1988)
- [4] Bickel, P.J., Li, B., et al.: Local polynomial regression on unknown manifolds. In: *Complex Datasets and Inverse Problems*, pp. 177–186. Institute of Mathematical Statistics (2007)
- [5] Binev, P., Dahmen, W., Lamby, P.: Fast high-dimensional approximation with sparse occupancy trees. *Journal of Computational and Applied Mathematics* **235**(8), 2063–2076 (2011)
- [6] Buchet, M., Chazal, F., Dey, T.K., Fan, F., Oudot, S.Y., Wang, Y.: Topological analysis of scalar fields with outliers. *arXiv preprint arXiv:1412.1680* (2014)
- [7] Buhmann, M.D.: *Radial Basis Functions: Theory and Implementations*, vol. 12. Cambridge University Press, Cambridge Monographs on Applied and Computational Science (2003)
- [8] Buhmann, M.D., De Marchi, S., Perracchione, E.: Analysis of a new class of rational RBF expansions. *IMA Journal of Numerical Analysis* **40**(3), 1972–1993 (2020)
- [9] Chen, M., Jiang, H., Liao, W., Zhao, T.: Efficient approximation of deep ReLU networks for functions on low dimensional manifolds. In: *Advances in Neural Information Processing Systems*, pp. 8174–8184 (2019)
- [10] Coifman, R.R., Lafon, S., Lee, A.B., Maggioni, M., Nadler, B., Warner, F., Zucker, S.W.: Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps. *Proceedings of the National Academy of Sciences* **102**(21), 7426–7431 (2005)
- [11] Coifman, R.R., Maggioni, M.: Diffusion wavelets. *Applied and Computational Harmonic Analysis* **21**(1), 53–94 (2006)
- [12] Cox, T.F., Cox, M.A.: *Multidimensional Scaling*. Chapman and Hall, London (2000)

- [13] Demko, S., Moss, W.F., Smith, P.W.: Decay rates for inverses of band matrices. *Mathematics of computation* **43**(168), 491–499 (1984)
- [14] DeVore, R., Petrova, G., Wojtaszczyk, P.: Approximation of functions of few variables in high dimensions. *Constructive Approximation* **33**(1), 125–143 (2011)
- [15] Dyn, N., Levin, D.: Iterative solution of systems originating from integral equations and surface interpolation. *SIAM Journal on Numerical Analysis* **20**(2), 377–390 (1983)
- [16] Faigenbaum-Golovin, S., Levin, D.: Manifold repairing, reconstruction and denoising from scattered data in high-dimension. *arXiv preprint arXiv:2102.01750* (2021)
- [17] Faigenbaum-Golovin, S., Levin, D.: Manifold reconstruction and denoising from scattered data in high dimension. *Journal of Computational and Applied Mathematics* **421**, 114818 (2023)
- [18] Fisher, R.A.: The use of multiple measurements in taxonomic problems. *Annals of Eugenics* **7**(2), 179–188 (1936)
- [19] Grohs, P., Sprechler, M.: Projection-based quasiinterpolation in manifolds. *SAM Report* **23** (2013)
- [20] He, X., Niyogi, P.: Locality preserving projections. In: *Advances in Neural Information Processing Systems*, pp. 153–160 (2004)
- [21] Levin, D.: The approximation power of moving least-squares. *Mathematics of Computation* **67**(224), 1517–1531 (1998)
- [22] Levin, D.: Between moving least-squares and moving least- ℓ_1 . *BIT Numerical Mathematics* **55**(3), 781–796 (2015)
- [23] Li, L.: A new complexity bound for the least-squares problem. *Computers & Mathematics with Applications* **31**(12), 15–16 (1996)
- [24] Lin, T., Zha, H.: Riemannian manifold learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **30**(5), 796–809 (2008)
- [25] Lipman, Y., Cohen-Or, D., Levin, D., Tal-Ezer, H.: Parameterization-free projection for geometry reconstruction. In: *ACM Transactions on Graphics (TOG)*, vol. 26, p. 22. ACM (2007)
- [26] Pearson, K.: LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* **2**(11), 559–572 (1901)
- [27] Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. *Science* **290**(5500), 2323–2326 (2000)

- [28] Shaham, U., Cloninger, A., Coifman, R.R.: Provable approximation properties for deep neural networks. *Applied and Computational Harmonic Analysis* **44**(3), 537–557 (2018)
- [29] Sober, B., Aizenbud, Y., Levin, D.: Approximation of functions over manifolds: A moving least-squares approach. arXiv preprint arXiv:1711.00765 (2017)
- [30] Tenenbaum, J.B., De Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. *Science* **290**(5500), 2319–2323 (2000)
- [31] Vardi, Y., Zhang, C.H.: The multivariate L1-median and associated data depth. *Proceedings of the National Academy of Sciences* **97**(4), 1423–1426 (2000)
- [32] Woodruff, D.P., et al.: Sketching as a tool for numerical linear algebra. *Foundations and Trends in Theoretical Computer Science* **10**(1–2), 1–157 (2014)
- [33] Wu, Zong-min., Schaback, R.: Local error estimates for radial basis function interpolation of scattered data. *IMA journal of Numerical Analysis* **13**(1), 13–27 (1993)
- [34] Yoon, J.: Spectral approximation orders of radial basis function interpolation on the Sobolev space. *SIAM Journal on Mathematical Analysis* **33**(4), 946–958 (2001)



Shira Faigenbaum-Golovin
Department of Mathematics,
Duke University,
Durham, NC 27705,
shira.golovin@math.duke.edu

David Levin,
School of Mathematical Sciences,
Tel Aviv University,
Tel Aviv, Israel,
levin@tau.ac.il