Robust Subspace-Constrained Quadratic Models for Low-Dimensional Structure Learning

Zheng Zhai and Xiaohui Li

***Abstract*—In this paper, we propose a robust subspace-constrained quadratic model (SCQM) for learning low-dimensional structure from high-dimensional data. Building upon the subspace-constrained quadratic matrix factorization (SQMF) framework, the proposed model accommodates a broad class of noise distributions, including generalized Gaussian and radial Laplace models. This generalization enables reliable performance under both heavy-tailed and light-tailed noise, thereby substantially enhancing robustness across diverse data regimes. To efficiently address the resulting nonconvex optimization problem, we develop a gradient-based algorithm equipped with a backtracking line-search strategy that ensures stable and efficient convergence. In addition, we present a sensitivity analysis of the ℓ_p^p and ℓ_2 loss functions, elucidating their distinct behaviors under varying noise characteristics. Extensive numerical experiments corroborate the theoretical analysis and demonstrate that the proposed approach consistently outperforms existing methods in terms of robustness and reconstruction accuracy.**

***Index Terms*—robust, quadratic, subspace-constrained optimization, manifold learning**

I. Introduction

LEARNING low-dimensional structures from high-dimensional data remains a fundamental challenge in data analysis, machine learning, and signal processing. Traditional methods, including local linear fitting [1], kernel density estimation (KDE) [2], linear matrix factorization, manifold fitting [3] and principal curves and surfaces [4], typically rely on the assumption that data reside in a linear subspace and are perturbed by Gaussian noise. While these assumptions enable the development of efficient algorithms, they are often unrealistic in real-world scenarios. In practice, data may lie on complex nonlinear manifolds and be affected by heavy-tailed noise or outliers.

The assumption of flatness often leads to focusing on small, localized regions of the data. However, when narrowing the focus to such regions, there may be insufficient samples to apply these algorithms effectively, resulting in biased estimates. This limitation can cause a significant loss of information and degrade model performance. To overcome these challenges, it is imperative to develop models that can accommodate a broader range of data structures, capturing the intricate, nonlinear nature of real-world data while remaining robust to noise and outliers. Such models will be better equipped to provide more accurate, generalizable results and offer enhanced performance in practical applications.

Zheng Zhai is with Department of Statistics, Faculty of Arts and Sciences at Beijing Normal University, Zhuhai. Xiaohui Li is with School of Mathematics and Information Sciences, Yantai University.

This work was supported by the National Natural Science Foundation of China (NSFC) under Grants of 12301478 and 11801592.

To address this limitation, manifold learning and nonlinear factorization models have been proposed. Among them, quadratic matrix factorization (QMF) [5] and subspace-constrained quadratic matrix factorization (SQMF) [6] have gained attention for their ability to incorporate second-order information and approximate curved manifolds. By augmenting linear subspace models [7], [8] with quadratic terms, these methods can capture both tangent directions and curvature, improving reconstruction accuracy and interpretability. In particular, subspace-constrained quadratic matrix factorization (SQMF) provides a principled framework for learning low-dimensional tangent and normal spaces, as well as a quadratic mapping between them.

Existing quadratic factorization models often rely on squared Euclidean loss or the Frobenius norm, both of which assume Gaussian noise. While these assumptions offer computational simplicity, they make the models highly sensitive to outliers and misspecifications. To mitigate the impact of outliers, typical quadratic factorization models introduce a penalty term that penalizes the contribution of the quadratic term, aiming to prevent overfitting. However, this approach introduces its own challenges, such as the difficult task of selecting appropriate penalty parameters.

Another key reason why the Frobenius norm loss may be inappropriate is due to the mismatch between its assumptions and the characteristics of real-world noise distributions. In practical applications—such as image analysis, sensor data processing, and robust representation learning—data often contains noise that is heavy-tailed, sparse, or impulsive. These types of noise can severely degrade model performance and lead to biased estimates. The assumption of Gaussian noise, which is commonly used in the Frobenius norm loss, fails to capture the true nature of noise in many real-world scenarios. This discrepancy highlights the necessity for more flexible and robust loss functions within quadratic factorization models, ones that are specifically designed to account for the diverse and complex noise patterns encountered in practice. Such loss functions would improve the model's ability to handle these noise complexities, ultimately enhancing performance and robustness in real-world applications.

In response to these challenges, we propose a robust subspace-constrained quadratic learning framework that allows for a broader class of noise distributions, including generalized Gaussian and Laplace families. This flexibility enables the model to adapt to both light-tailed and heavy-tailed noise while maintaining the expressive power of quadratic manifold representations. Consequently, the proposed model offers enhanced robustness and accuracy in practical scenarios where Gaussian

assumptions fail.

From an optimization perspective, we extend the classical Frobenius-norm formulation $\|\cdot\|_{\mathrm{F}}^2$ to matrix factorization under alternative norms, such as the entrywise $\ell_{1,1}$ norm and the mixed $\ell_{2,1}$ norm. To solve the resulting optimization problem, we propose a gradient descent-based approach. Due to the quadratic structure of the mapping and the orthogonality constraints, the optimization remains nonconvex. To address this, we derive the Karush-Kuhn-Tucker (KKT) conditions and develop a Riemannian gradient descent algorithm on the Stiefel manifold, ensuring feasibility and convergence through orthogonality-preserving updates.

The main contributions of this work are as follows:

- We propose a generalized subspace-constrained quadratic framework that extends existing SQMF models to a broad class of loss functions and derive explicit gradients for all variables, including latent coordinates.
- We provide a theoretical analysis, based on the implicit function theorem, that explains why the ℓ_p^p loss and non-squared Euclidean norm improve robustness.
- We develop an efficient Riemannian gradient descent algorithm with orthogonality-preserving updates on the Stiefel manifold.
- Extensive numerical experiments demonstrate that the proposed method significantly improves robustness and reconstruction accuracy in the presence of outliers.

The structure of the paper is as follows. In Section II, we introduce the subspace-constrained quadratic model, detailing its formulations and the associated identifiability property. We also provide a toy example featuring a circle in $\mathbb{R}^2$ to demonstrate the model's effectiveness. Section III focuses on deriving the gradients with respect to the free variables and orthogonal constraints, presenting the KKT conditions, and outlining the proposed optimization algorithm. In Section IV, we conduct a convexity analysis of the subproblems related to c, Θ, and the projection with respect to τ. Section V offers a sensitivity analysis for the ℓ_p^p and ℓ_2 loss functions. Section VII presents numerical experiments to validate the theoretical results, and the paper concludes in Section VIII with a discussion of promising directions for future research.

A. Related works

Since our work leverages a subspace-constrained quadratic matrix factorization framework to learn latent manifold structures, it is closely related to a broad class of manifold fitting and denoising methods. These approaches aim to recover low-dimensional geometric structures embedded in high-dimensional observations, typically under noise and sampling irregularities. Representative techniques include local linear fitting methods such as local principal component analysis (LPCA) [1], ridge-based approaches derived from kernel density estimation (KDE) [9] and its logarithmic variant (LOG-KDE) [10], tangent-space aggregation methods such as manifold fitting (MFIT) [11], and higher-order regression techniques including moving least squares (MLS) [12]. For a comprehensive comparison, we focus on seven representative methods spanning these categories.

Ridge estimation constitutes one of the most influential paradigms for nonparametric manifold estimation. Rather than explicitly parameterizing the manifold, ridge-based methods define it implicitly as a set of points satisfying specific differential conditions of an estimated probability density function. In particular, KDE-based ridge estimation constructs a smooth density estimate from the observed data and identifies the manifold as a subset where the gradient aligns with the principal eigenspace of the Hessian, while the remaining curvature directions exhibit concavity. This formulation allows the manifold to be recovered directly from data geometry without requiring an explicit embedding or coordinate chart. In practice, the ridge set rarely admits a closed-form solution, and iterative procedures such as the subspace-constrained mean shift (SCMS) [4], [13] algorithm are commonly employed to trace the ridge structure. The LOG-KDE variant further modifies this framework by applying the ridge conditions to the logarithm of the density, leading to a different curvature characterization and tangent space estimation, often improving robustness in regions of varying density.

Beyond ridge-based techniques, several methods reconstruct manifolds by directly exploiting local tangent or normal space information. A prominent example is manifold fitting (MFIT), which estimates the manifold by enforcing consistency across locally estimated normal spaces. MFIT aggregates normal directions obtained from neighborhoods of nearby points using spatially adaptive weights and defines the manifold as the set of points where the weighted normal projections vanish. This approach provides a principled way to fuse local geometric information into a globally coherent manifold estimate and is particularly effective when normal directions can be reliably estimated from noisy data.

Higher-order manifold estimation methods aim to move beyond linear or first-order approximations by explicitly modeling local curvature. Among these, the moving least squares (MLS) framework is one of the most widely studied. MLS first estimates a local tangent space around each query point and then performs a weighted polynomial regression in this local coordinate system. The fitted polynomial captures higher-order geometric features, and the manifold estimate is obtained by projecting the query point onto the polynomial surface. Owing to its flexibility and strong approximation properties, MLS is well suited for smooth manifolds with nontrivial curvature, albeit at the cost of increased computational complexity and sensitivity to parameter choices.

Another closely related line of work extends local linear models by incorporating explicit geometric primitives. Spherical PCA (SPH), also known as spherelets, augments local PCA [1] by fitting low-dimensional spherical structures instead of affine subspaces. The method first projects data into a locally estimated affine subspace of dimension one higher than the intrinsic dimension and then fits a sphere within this space. By modeling curvature through spherical geometry, SPH provides a more accurate approximation than linear methods when the underlying manifold exhibits approximately constant curvature.

In contrast to the above approaches, which primarily rely on local geometric estimation and projection-based recon-

struction, our method adopts the generalized quadratic approximation perspective with explicit subspace constraints. By integrating quadratic structure and subspace regularization, the proposed framework bridges manifold learning and data approximation, enabling robust recovery of latent manifolds while maintaining computational efficiency and scalability. This distinction allows our approach to complement existing local fitting and denoising methods, particularly in settings where global structure and latent representations are of primary interest.

II. SCQM AND ITS FORMULATIONS

In this section, we present the generalized SCQM model and explore various loss functions, highlighting their appropriate use cases and scenarios. We also provide a toy example to demonstrate the advantages of SCQM over the traditional linear local fitting model, illustrating how SCQM model can better capture complex patterns and improve performance in practical applications.

A. Quadratic Fitting Model with Subspace Constraint

We generalize classical quadratic matrix factorization beyond the Frobenius-norm objective by allowing a broad class of loss functions. This enables estimation in models of the form

$$x_i = f(\tau_i) + \epsilon_i, \qquad i = 1, \dots, n, \tag{1}$$

where the noise term ϵ_i is not restricted to be Gaussian. Instead, we allow ϵ_i to follow either heavy-tailed or light-tailed distributions. A representative example is the generalized Gaussian distribution with density $p(\epsilon) \propto \exp\big(-\|\epsilon\|_p^p/\eta\big),\ x \in \mathbb{R}^D$, which includes the Gaussian model when $p = 2$ and the Laplace model when $p = 1$. We also consider isotropic, rotation-invariant noise modeled by the multivariate radial Laplace distribution $p(\epsilon) \propto \exp(-\|\epsilon\|_2/\eta)$.

When the noise distribution is known, maximum likelihood estimation of the mapping $f(\cdot)$ reduces, up to additive constants, to the optimization problem

$$\min_{f,\{\tau_i\}} \sum_{i=1}^{n} \ell\big(x_i - f(\tau_i)\big), \tag{2}$$

where $\ell(\cdot)$ is the negative log-likelihood associated with the assumed noise model. Typical choices include $\|r\|_p^p$, $\|r\|_2$, and $\|r\|_2^2$, corresponding to generalized Gaussian, radial Laplace, and Gaussian noise models, respectively. This formulation establishes a direct link between the loss function and the statistical properties of the noise.

In what follows, we study the solution to (2) by explicitly specifying both the loss function $\ell(\cdot)$ and the function class of f. In particular, we restrict f to the class of quadratic functions equipped with explicit tangent- and normal-space basis representations. This choice offers two key advantages. First, it provides a clear geometric interpretation of the model parameters while retaining strong representation capability. Second, the linear model naturally arises as a special case of this framework by setting all quadratic terms to zero. As a result, this formulation allows us to seamlessly degenerate the quadratic model into its linear counterpart, thereby facilitating a direct and systematic investigation of the effectiveness and necessity of the quadratic terms.

1) Quadratic Function Class: To improve the quality of approximation over a broader domain, we restrict $f(\cdot)$ to the class of quadratic mappings for two main reasons. First, polynomial parameterizations offer significant practical advantages in optimization, as they lead to smooth objectives with well-structured gradients and Hessians. Second, the quadratic model is consistent with the manifold assumption, as it explicitly captures the interaction between the tangent space and the normal space through second-order terms. Specifically, we define the restricted class of quadratic mappings $f(\cdot)$ as follows:

$$\begin{aligned}\mathcal{F} = \big\{ f : \mathbb{R}^d \to \mathbb{R}^D \;\big|\; & f(\tau) = c + U\tau + V\,\mathcal{A}(\tau,\tau), \\ & U^\top U = I_d,\ V^\top V = I_s,\ U^\top V = 0 \big\}.\end{aligned}$$

Here, c denotes the shift parameter. The columns of U form an orthonormal basis of the tangent space, while the columns of V span a subspace orthogonal to that of U. The vector $\mathcal{A}(\tau,\tau) \in \mathbb{R}^s$ represents the action of a third-order tensor $\mathcal{A}$ on (τ,τ), defined element-wise by

$$\big\{\mathcal{A}(\tau,\tau)\big\}_k = \sum_{i,j=1}^{d} \mathcal{A}_{k,i,j}\,\tau_i\tau_j, \qquad k = 1, \dots, s.$$

Since the rank-one matrix $\tau\tau^\top$ is symmetric, we may assume without loss of generality that each slice $\mathcal{A}_k \in \mathbb{R}^{d\times d}$ is symmetric. By concatenating U and V into a single matrix $Q = [U, V] \in \mathbb{R}^{D\times(d+s)}$, the orthogonality constraints can be written compactly as $Q^\top Q = I_{d+s}$. This yields the equivalent representation

$$f(\tau) = c + Q\begin{bmatrix}\tau \\ \mathcal{A}(\tau,\tau)\end{bmatrix}, \qquad Q^\top Q = I_{d+s}. \tag{3}$$

Exploiting the symmetry of $\mathcal{A}$, there exists a matrix Θ such that $\mathcal{A}(\tau,\tau) = \Theta^\top \mathrm{vech}(\tau\tau^\top)$, where $\mathrm{vech}(\cdot)$ stacks the upper-triangular entries of a symmetric matrix into a vector. Consequently, $\mathcal{F}$ admits the equivalent formulation

$$f(\tau) = c + Q\begin{bmatrix}\tau \\ \Theta^\top \mathrm{vech}(\tau\tau^\top)\end{bmatrix}, \qquad Q^\top Q = I_{d+s}. \tag{4}$$

Our learning problem is:

$$(\widehat{f},\ \{\widehat{\tau}_i\}) = \arg\min_{f\in\mathcal{F},\,\{\tau_i\}} \sum_{i=1}^{n} \ell\big(x_i - f(\tau_i)\big). \tag{5}$$

This model can be viewed as a generalized formulation of the subspace-constrained quadratic factorization problem studied in [6], in which the Frobenius norm is replaced by a column-wise loss defined through a specialized norm.

From Eqn. (5), we observe that the model simultaneously learns both the local geometry, represented by the parameters c, Q, and Θ in the function f, as well as the coordinates $\{\tau_i\}$ derived from the high-dimensional input data $\{x_i\}$. Once we obtain $\widehat{f}$, we can use it as a model to refine the new noisy data y by projecting it onto $\widehat{f}$, which is achieved by solving the following optimization problem:

$$\widehat{y} = \arg\min_{x=\widehat{f}(\tau)} \ell(x - y). \tag{6}$$

From an algorithmic perspective, due to the nonlinearity of both the quadratic mapping and the general loss function, closed-form solutions are generally not available for the optimization problem in Eqn. (5) and Eqn. (6). In the following sections, we first address the identifiability issue and discuss principled criteria for selecting an appropriate loss function. We then propose an efficient gradient-based algorithm to solve the resulting optimization problem numerically.

B. Identifiability

The optimization problem in Eqn. (5) is not identifiable due to the inherent invariances in the latent representation. Specifically, the model exhibits invariance under orthogonal transformations of the latent variables.

Let $R \in O(d)$ be any orthogonal matrix, and define the transformed latent variable $\eta = R\tau$. Since

$$\eta\eta^\top = R(\tau\tau^\top)R^\top,$$

there exists a matrix $S(R)$ such that $\mathrm{vech}(\eta\eta^\top) = S(R)\,\mathrm{vech}(\tau\tau^\top)$. Consequently, by defining $U_R = UR^T, \Theta_R = S(R)^{-\top}\Theta$, we obtain

$$\Theta^\top \mathrm{vech}(\tau\tau^\top) = \Theta_R^\top \mathrm{vech}(\eta\eta^\top), \qquad U\tau = U_R\eta.$$

This implies that different parameter tuples (c, U, V, Θ, τ) and $(c, U_R, V, \Theta_R, \eta)$, related by orthogonal transformations $\eta = R\tau$, induce the same input-output mapping $f(\tau) = f_R(\eta)$, where $f_R(\cdot)$ denotes the function parameterized by c, U_R, V, Θ_R.

Therefore, the parameters of the quadratic factorization model are identifiable only up to equivalence classes induced by orthogonal transformations in the latent space. Although a unique parameterization cannot be recovered, the learned mapping $f(\cdot)$ and the associated quadratic manifold are uniquely determined within these equivalence classes.

C. Loss functions

In this section, we investigate how to choose an appropriate loss function for the quadratic factorization model. The central principle is that the loss function should be aligned with the statistical distribution of the noise in the data or observations. Specifically, when the noise exhibits a long-tailed (heavy-tailed) distribution, an ℓ_p^p loss with $p < 2$ provides a more robust and suitable choice. In contrast, when the noise follows a short-tailed distribution, the Gaussian assumption becomes appropriate, and the corresponding ℓ_2^2 loss is a natural and effective option. This distribution-aware selection enables the model to better capture the underlying data characteristics and achieve improved robustness and accuracy.

D. Criteria for Choosing the Loss Function

We discuss the criteria for loss-function selection under three different scenarios. First, we consider the simplest setting, in which the noise distribution ϵ_i is known *a priori*. Second, we study the case where the contaminated noise samples ϵ_i are directly observable. Third, we address the most realistic scenario, in which the noise is implicitly embedded in the observations and cannot be isolated from the underlying signal.

a) Known noise distribution: In the first scenario, we select the loss function $\ell(\cdot)$ according to a principled, likelihood-based criterion under known noise assumptions. By the law of large numbers, as the sample size increases, the empirical distribution of the noise converges to its true underlying distribution. Consequently, a statistically sound and natural choice of the loss function is the negative log-likelihood induced by the assumed noise model. This choice leads to statistically consistent estimators and, with high probability, ensures that the optimization objective faithfully reflects the intrinsic characteristics of the noise. Guided by this principle, we consider several commonly adopted noise models together with their corresponding loss functions in the following analysis.

When the noise is isotropic Gaussian, i.e., $\epsilon_i \sim \mathcal{N}(0, \sigma^2 I)$, the negative log-likelihood is proportional to the squared Euclidean norm. Accordingly, we adopt the loss $\ell(r) = \|r\|_2^2$, which recovers the classical least-squares formulation. If the noise instead follows a multivariate isotropic (radial) Laplace distribution with density $p(\epsilon) \propto \exp(-\lambda\|\epsilon\|_2)$, the corresponding negative log-likelihood leads to the Euclidean norm loss $\ell(r) = \|r\|_2$. When the noise components are independently distributed according to univariate Laplace laws, the negative log-likelihood becomes proportional to the ℓ_1 norm, and we employ $\ell(r) = \|r\|_1$, which is well known for its robustness to sparse, large-magnitude outliers.

More generally, heavy-tailed or light-tailed noise can be modeled by a generalized Gaussian distribution with density $p(\epsilon) \propto \exp(-|\epsilon|_p^p)$ for $p \geq 1$, motivating the loss $\ell(r) = \|r\|_p^p$. Smaller values of p yield increased robustness to outliers, while larger values recover the Gaussian setting. When the noise covariance is anisotropic, i.e., $\epsilon_i \sim \mathcal{N}(0, \Sigma)$ with $\Sigma \succ 0$, the appropriate choice is the squared Mahalanobis loss $\ell(r) = r^\top \Sigma^{-1} r$, which accounts for correlations and heterogeneous scaling across dimensions. Finally, in the presence of mixed noise models or outlier contamination, the Huber loss is frequently adopted, as it interpolates smoothly between the ℓ_2 and ℓ_1 norms, combining robustness with local smoothness.

Table I summarizes several commonly used loss functions together with their associated noise models from a maximum-likelihood perspective. Overall, the choice of the loss function $\ell(\cdot)$ plays a central role in balancing statistical efficiency and robustness, and is therefore critical to the performance of the proposed quadratic factorization framework.

b) Observable noise samples: In the second scenario, where the noise samples $\{\epsilon_i\}_{i=1}^n$ are explicitly observable, the optimal shape parameter p in the ℓ_p^p loss can be estimated via maximum likelihood estimation (MLE) [14]. Since no closed-form solution is available, we suggest to employ a numerical optimization procedure based on the profile likelihood. In particular, a one-dimensional search strategy, such as grid search or bisection, is adopted, following standard practice in the literature [15], [16].

c) Unobservable noise embedded in observations.: In the third and most practical scenario, the noise is intrinsically mixed with the unknown underlying signal, rendering direct maximum likelihood estimation of the noise distribution infeasible. Nevertheless, meaningful and practically effective

TABLE I
COMMON LOSS FUNCTIONS, TOGETHER WITH THEIR GRADIENTS AND HESSIANS WITH RESPECT TO x, AND THE CORRESPONDING NOISE DISTRIBUTIONS UNDER A MAXIMUM LIKELIHOOD INTERPRETATION.

Loss function	$\boldsymbol{\ell(x-y)}$	$\boldsymbol{\nabla_x \ell(x-y)}$	$\boldsymbol{\nabla_x^2 \ell(x-y)}$	**Suitable Distribution**
ℓ_1 (Manhattan)	$\sum_{k=1}^{d} \lvert x_k - y_k \rvert$	$\operatorname{sign}(x-y)$ *(subgradient)*	0 a.e. *(undefined at $x=y$)*	Independent Laplace
ℓ_2 (Euclidean)	$\lVert x-y \rVert_2$	$\dfrac{x-y}{\lVert x-y \rVert_2}, \quad x \neq y$	$\dfrac{1}{\lVert x-y \rVert_2}\left(I - \dfrac{(x-y)(x-y)^T}{\lVert x-y \rVert_2^2}\right)$	Multivariate (isotropic) Laplace
ℓ_2^2 (Squared Euclidean)	$\lVert x-y \rVert_2^2$	$2(x-y)$	$2I$	Multivariate Gaussian
ℓ_p^p $(p \geq 1)$	$\sum_{k=1}^{d} \lvert x_k - y_k \rvert^p$	$p\,\lvert x-y \rvert^{p-1} \odot \operatorname{sign}(x-y)$	$p(p-1)\,\operatorname{diag}\big(\lvert x-y \rvert^{p-2}\big)$	Generalized Gaussian
Mahalanobis (squared)	$(x-y)^T M (x-y)$	$2M(x-y)$	$2M$	Multivariate Gaussian
Huber distance	$\begin{cases} \frac{1}{2}\lVert x-y \rVert_2^2, & \lVert x-y \rVert_2 \leq \delta, \\ \delta \lVert x-y \rVert_2 - \frac{1}{2}\delta^2, & \lVert x-y \rVert_2 > \delta \end{cases}$	$\begin{cases} x-y, & \lVert x-y \rVert_2 \leq \delta, \\ \delta \dfrac{x-y}{\lVert x-y \rVert_2}, & \lVert x-y \rVert_2 > \delta \end{cases}$	$\begin{cases} I, & \lVert x-y \rVert_2 \leq \delta, \\ \delta\, \nabla^2 \lVert x-y \rVert_2, & \lVert x-y \rVert_2 > \delta \end{cases}$	Gaussian–Laplace hybrid

criteria can still be employed for loss-function selection. In this case, we propose the following strategies:

1) **Control of heavy-tailed noise** Employing the ℓ_p^p loss with a relatively small value of p mitigates the influence of heavy-tailed noise. This choice reduces sensitivity to samples that lie far from the estimated curve, thereby enhancing robustness to large deviations.
2) **Euclidean-distance-based projection.** Both the ℓ_2^2 and ℓ_2 losses yield projections that minimize the Euclidean distance between data points and the estimated model, aligning with the classical notion of projection. In particular, the ℓ_2^2 loss corresponds to the maximum likelihood estimator under Gaussian noise, which is widely adopted in practice.
3) **A conservative choice.** The ℓ_2 loss serves as a conservative and stable choice, as it exhibits reliable performance across a wide range of noise conditions, including light-tailed and moderately heavy-tailed distributions. Moreover, when computing the optimal local coordinates associated with the fitted low-dimensional surface (or curve in $\mathbb{R}^2$), the ℓ_2 and ℓ_2^2 losses are equivalent in the sense that they induce the same minimizers. Consequently, both losses admit the standard distance-based interpretation of orthogonal projection, which is consistent with classical geometric formulations.

E. A toy example in $\mathbb{R}^2$

In this experiment, we demonstrate that when the loss function is chosen to match the underlying noise distribution, all models achieve strong performance. We present a synthetic example demonstrating that the choice of loss function should be consistent with the underlying noise distribution. Specifically, we generate data according to

$$x_i = [\cos(t_i), \sin(t_i)]^T + \epsilon_i, \qquad \epsilon_i \sim C_{p,d} \exp\big(-\|\epsilon_i\|_p^p\big), \tag{7}$$

where $C_{p,d}$ denotes the normalizing constant, $\{t_i\}$ consists of equally spaced points in the interval $[0, 4]$, and ϵ_i follows a multivariate generalized Gaussian distribution with shape parameter p. Also, to assess the performance of the ℓ_2 loss under a matched noise model, we additionally consider a setting in which the noise follows a multivariate isotropic Laplace distribution, namely, $\epsilon_i \sim C_d' \exp\big(-\|\epsilon_i\|_2\big)$, which corresponds to the negative log-likelihood associated with the Euclidean norm.

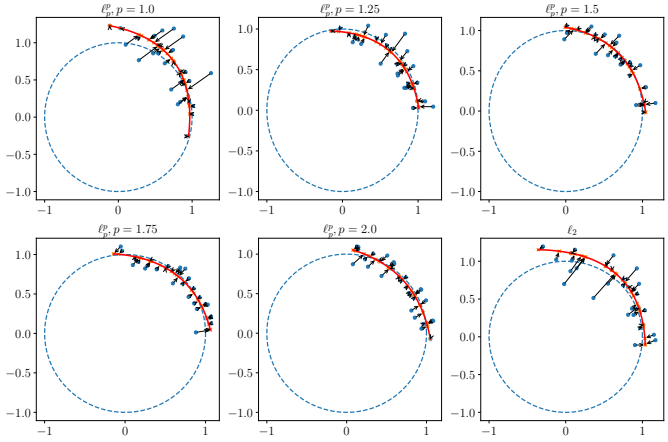

Fig. 1. Illustration of the fitted curves and projection points obtained using ℓ_p^p losses with different values of p.

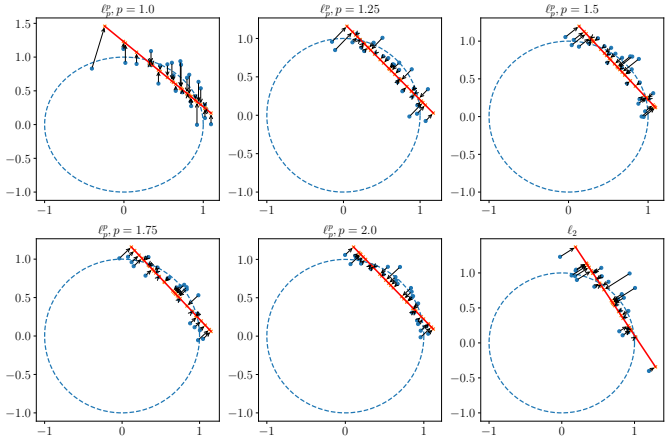

Fig. 2. Illustration of the fitted curves and projection points obtained using ℓ_p^p losses with different values of p when restricting the function class to be linear ($\mathcal{A} = \mathbf{0}$).

To this end, we describe how to generate synthetic noise samples whose density is proportional to $\exp(-\|\epsilon\|_p^p)$. We fo-

cus on the multivariate generalized Gaussian distribution [17], whose probability density function takes exactly this form. Owing to its separability, the distribution factorizes into independent generalized Gaussian marginals along each coordinate. Consequently, a random vector in $\mathbb{R}^d$ can be generated by independently sampling each coordinate from the one-dimensional density $f(t) \propto \exp(-|t|^p)$. In practice, such samples can be efficiently constructed using a simple transformation: draw $u \sim \mathrm{Gamma}(1/p, 1)$ and an independent Rademacher random variable $s \in \{-1, 1\}$ with equal probability, and set $\epsilon = s\,u^{1/p}$. Repeating this procedure independently for each coordinate yields a random vector whose joint distribution follows the multivariate generalized Gaussian distribution.

In Fig. 1, the fitted model f is shown by the red curve, while the blue points denote the noisy observations. The arrows indicate the orthogonal projections of the noisy data points onto the underlying low-dimensional quadratic manifold defined by f. The noisy data are generated under different values of the shape parameter $p \in \{1.0, 1.25, 1.5, 1.75, 2.0\}$, and the corresponding $\|\cdot\|_p^p$ loss is employed in each case.

We make the following observations. First, the quadratic subspace-constrained model exhibits substantially stronger fitting capability than its linear counterpart, achieving tighter approximation over a broader domain. Second, the underlying low-dimensional structure can be successfully recovered by the proposed subspace-constrained quadratic matrix factorization model, provided that the loss function is properly matched to the noise distribution. In particular, for heavy-tailed noise, we recommend using the ℓ_p^p loss with smaller values of p to enhance robustness, whereas for light-tailed noise, larger values of p, such as the squared Euclidean loss, are more suitable and yield higher statistical efficiency.

III. Gradients and KKT condition

In this section, we derive closed-form expressions for $\nabla_\tau F$, $\nabla_\Theta F, \nabla_c F$, and $\nabla_Q F$. The gradient $\nabla_\tau F$ is nontrivial to compute due to the involvement of the element-extraction operator $\mathrm{vech}(\cdot)$.

A. Gradient with respect to $\{\tau_k\}_{k=1}^n$

Here, we derive the gradient of the loss function with respect to the latent variable τ_k. Owing to the separability of the loss across data samples, the objective function can be decomposed into independent terms, each depending only on a single latent representation. Consequently, the gradient with respect to τ_k can be computed by differentiating only the k-th loss term.

$$\nabla_{\tau_k} F = \nabla_{\tau_k} \sum_{i=1}^{n} \ell(x_i - f(\tau_i)) = \nabla_{\tau_k} \ell(x_k - f(\tau_k)).$$

Due to the nonlinear dependence of $f(\tau_k)$ on τ_k through $\mathrm{vech}(\tau_k \tau_k^T)$, a direct computation of $\nabla_{\tau_k} f(\tau_k)$ is nontrivial. Observing that, for a small perturbation $\delta \in \mathbb{R}^d$, the function $f(\tau_k + \delta)$ admits the following first-order expansion:

$$f(\tau_k + \delta) = f(\tau_k) + \langle \nabla_{\tau_k} f(\tau_k), \delta \rangle + \mathcal{O}(\|\delta\|_2^2). \tag{8}$$

Based on (8), we derive the gradient by analyzing the difference $f(\tau_k + \delta) - f(\tau_k)$ by

$$\begin{aligned} &f(\tau_k + \delta) - f(\tau_k) \\ =&U\delta + V\Theta^T \mathrm{vech}((\tau_k + \delta)(\tau_k + \delta)^T) + V\Theta^T \mathrm{vech}(\tau_k \tau_k^T) \\ =&U\delta + V\Theta^T \mathrm{vech}(\tau_k \delta^T + \delta \tau_k^T + \delta\delta^T). \end{aligned}$$

Define the operators $T_\tau(\delta) = \mathrm{vech}(\tau\delta^T)$ and $T'_\tau(\delta) = \mathrm{vech}(\delta\tau^T)$. Observe that both $\mathrm{vech}(\tau\delta^T)$ and $\mathrm{vech}(\delta\tau^T)$ induce linear mappings with respect to the perturbation δ. In particular, for any $\delta_1, \delta_2 \in \mathbb{R}^d$ and any scalar $\kappa \in \mathbb{R}$, these operators satisfy the additivity and homogeneity properties,

$$T_{\tau_k}(\kappa\delta) = \mathrm{vech}\big(\tau_k(\kappa\delta)^T\big) = \kappa \mathrm{vech}(\tau_k \delta^T) = \kappa T_{\tau_k}(\delta).$$

$$\begin{aligned} T_{\tau_k}(\delta_1 + \delta_2) =&\mathrm{vech}\big(\tau_k(\delta_1 + \delta_2)^T\big) \\ =&\mathrm{vech}(\tau_k \delta_1^T) + \mathrm{vech}(\tau_k \delta_2^T) \\ =&T_{\tau_k}(\delta_1) + T_{\tau_k}(\delta_2). \end{aligned}$$

Therefore, both T_{τ_k} and T'_{τ_k} are linear mappings from $\mathbb{R}^d$ to $\mathbb{R}^{\frac{d(d+1)}{2}}$. By the matrix representation theorem for linear transformations [18], any linear operator admits a matrix representation once the input and output bases are fixed. In the following, we adopt the standard bases and derive the explicit matrix form of the corresponding linear transformation.

Corollary 1 (Matrix representation theorem) *Let $T : \mathbb{R}^n \to \mathbb{R}^m$ be a linear transformation. Then there exists a unique matrix $A \in \mathbb{R}^{m \times n}$ such that*

$$T(x) = Ax, \quad \forall x \in \mathbb{R}^n.$$

In order to find the corresponding transformation matrix. We take the standard basis of $\{e_1, e_2, ..., e_d\} \in \mathbb{R}^d$, Then, any $\delta \in \mathbb{R}^d$ can be written as

$$\begin{aligned} T_{\tau_k}(\delta) =&T_{\tau_k}(\sum_{i=1}^{d} \langle \delta, e_i \rangle e_i) = \sum_{i=1}^{d} \langle \delta, e_i \rangle T_{\tau_k}(e_i) \\ =&[T_{\tau_k}(e_1), T_{\tau_k}(e_2), ..., T_{\tau_k}(e_d)]\delta. \end{aligned}$$

Therefore, by defining $M_{\tau_k} = [T_{\tau_k}(e_1), T_{\tau_k}(e_2), ..., T_{\tau_k}(e_d)]$ and $N_{\tau_k} = [T'_{\tau_k}(e_1), T'_{\tau_k}(e_2), ..., T'_{\tau_k}(e_d)]$, we have:

$$f(\tau_k + \delta) - f(\tau_k) = U\delta + V\Theta^T(M_{\tau_k} + N_{\tau_k})\delta + V\Theta^T \mathrm{vech}(\delta\delta^T).$$

By the definitions of the linear operators $T(\cdot)$ and $T'(\cdot)$, the explicit forms of $T_{\tau_k}(e_i)$ and $T'_{\tau_k}(e_i)$ can be readily derived for all $i = 1, \ldots, d$. Neglecting the higher-order term $V\Theta^T \mathrm{vech}(\delta\delta^T)$ in the first-order expansion, we obtain $\nabla_\tau f(\tau) = U + V\Theta^T(M_\tau + N_\tau)$. Therefore, the gradient $\nabla_\tau F$ is:

$$\begin{aligned} \nabla_{\tau_k} F(c, Q, \Theta, \Phi) =&\nabla_{\tau_k} \ell(x_k - f(\tau_k)) \\ =&\nabla_{\tau_k}^T f(\tau_k) \nabla_y \ell(y - x_k)|_{y = f(\tau_k)}. \end{aligned} \tag{9}$$

Note that the Jacobian matrix $\nabla_{\tau_k} f(\tau_k)$ is defined with respect to the k-th latent representation τ_k corresponding to x_k. In the following, we present the general forms of the matrices M_τ and N_τ, which characterize the linear mappings induced by the quadratic term. By stacking the vectors $T_\tau(e_j), j = 1, \ldots, d$, as the columns of $M_\tau \in \mathbb{R}^{\{(d^2+d)/2\} \times d}$, and $T'_\tau(e_j), j = 1, \ldots, d$, as the columns of $N_\tau \in \mathbb{R}^{\{(d^2+d)/2\} \times d}$,

we obtain explicit matrix representations of the two linear operators. Consequently, for any $\tau \in \mathbb{R}^d$, the transformation matrices M_τ and N_τ admit the following structured forms:

$$M_\tau = \begin{pmatrix} \tau_{[1]} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \tau_{[1]} & 0 & 0 & 0 & 0 & 0 \\ & & \cdots & \cdots & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & \tau_{[1]} \\ 0 & \tau_{[2]} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \tau_{[2]} & 0 & 0 & 0 & 0 \\ & & \cdots & \cdots & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & \tau_{[2]} \\ & & \cdots & \cdots & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & \tau_{[d]} \end{pmatrix},$$

$$N_\tau = \begin{pmatrix} \tau_{[1]} & 0 & 0 & 0 & 0 & 0 & 0 \\ \tau_{[2]} & 0 & 0 & 0 & 0 & 0 & 0 \\ & & \cdots & \cdots & & & \\ \tau_{[d]} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \tau_{[2]} & 0 & 0 & 0 & 0 & 0 \\ 0 & \tau_{[3]} & 0 & 0 & 0 & 0 & 0 \\ & & \cdots & \cdots & & & \\ 0 & \tau_{[d]} & 0 & 0 & 0 & 0 & 0 \\ & & \cdots & \cdots & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & \tau_{[d]} \end{pmatrix}.$$

Noting that both M_τ and N_τ are highly sparse matrices, each containing exactly one nonzero entry per row, we can express them in the following compact form:

$$N_\tau = [\tau_{[1]}e_1, \tau_{[2]}e_1, \cdots, \tau_{[d]}e_1, \tau_{[2]}e_2, \cdots, \tau_{[d]}e_2, \cdots, \tau_d e_d]^T,$$
$$M_\tau = [\tau_{[1]}e_1, \tau_{[1]}e_2, \cdots, \tau_{[1]}e_d, \tau_{[2]}e_2, \cdots, \tau_{[2]}e_d, \cdots, \tau_{[d]}e_d]^T.$$

Here, $\tau_{[k]} \in \mathbb{R}$ denotes the k-th scalar component of $\tau \in \mathbb{R}^d$, which is distinguished from $\tau_k \in \mathbb{R}^d$, the k-th latent vector associated with the data point $x_k \in \mathbb{R}^D$. The vector e_k denotes the k-th canonical basis (one-hot) vector, whose k-th entry is equal to 1 and all other entries are zero.

B. Riemannian Gradient and First-Order Optimality Condition for Q

In this subsection, we compute the Riemannian gradient G_Q and update Q by performing a gradient step along the tangent direction followed by a retraction onto the Stiefel manifold [19]:

$$\begin{cases} [\widetilde{Q}, R] = \mathrm{qr}\Big(Q^{(t)} - \eta_t G_{Q^{(t)}}\Big), \\ Q^{(t+1)} = \widetilde{Q}\,\mathrm{diag}\big(\mathrm{sign}(\mathrm{diag}(R))\big), \end{cases} \tag{10}$$

where $\eta_t > 0$ denotes the step size. Here, the QR decomposition acts as a retraction operator, mapping the updated iterate back onto the Stiefel manifold, while the diagonal sign correction ensures the uniqueness of the factorization and preserves continuity of the iterates. The Riemannian update on $\mathrm{St}(d+s, D)$

$$\begin{aligned} G_Q =& \nabla_Q F - Q^{\mathrm{sym}}\big(Q^T \nabla_Q F\big) \\ =& Q\frac{Q^T\nabla_Q F - \nabla_Q^T F Q}{2} + Q^{\perp}(Q^{\perp})^T)\nabla_Q F, \end{aligned}$$

where the Euclidean gradient $\nabla_Q F$ with a specific form by

$$\begin{aligned} \nabla_Q F(c, Q, \Theta, \Phi) =& \sum_{i=1}^n \nabla_Q \ell(f(\tau_i) - x_i) \\ =& \sum_{i=1}^n \nabla_y \ell(y)|_{y=f(\tau_i)-x_i} \left[\tau_i^T, \mathrm{vech}^T(\tau_i\tau_i^T)\Theta\right]. \end{aligned}$$

Proposition 1 *If the Riemannian gradient with respect to Q vanishes, i.e., $G_Q = \mathbf{0}$, then Q satisfies the first-order optimality condition*

$$\nabla_Q F(c, Q, \Theta, \Phi) - 2Q\Lambda = \mathbf{0}, \qquad Q^\top Q = I_{d+s}, \tag{11}$$

where $\Lambda \in \mathbb{R}^{(d+s)\times(d+s)}$ is a symmetric matrix of Lagrange multipliers.

The proof is placed in the appendix. It is worth noting that although the loss function is an affine mapping composed with a convex outer function, the resulting objective is not convex with respect to Q due to the nonconvex geometry induced by the Stiefel manifold constraint.

C. Gradient for Θ and c

Since the loss function involves linear transformations with respect to both c and Θ, followed by an outer norm operation, the optimization properties of c and Θ are inherently coupled. We therefore analyze these two variables jointly in this subsection.

a) Gradient with respect to Θ: For each i, the mapping $\Theta \mapsto V\Theta^T\mathrm{vech}(\tau_i\tau_i^T)$ is affine in Θ. Since the loss function $\ell(\cdot)$ is convex, the composition $\ell(f_{\Theta,c}(\tau_i) - x_i)$ is convex with respect to Θ. Applying the chain rule, we obtain

$$\begin{aligned} &\nabla_\Theta \sum_{i=1}^n \ell(f_{\Theta,c}(\tau_i) - x_i) \\ =& \sum_{i=1}^n \mathrm{vech}(\tau_i\tau_i^T)\Big(\nabla_y \ell(y - x_i)^T\Big)\Big|_{y=f_{\Theta,c}(\tau_i)} V. \end{aligned}$$

b) Gradient with respect to c: Similarly, the inner function is affine in c, and the gradient is given by

$$\nabla_c \sum_{i=1}^n \ell(f_{\Theta,c}(\tau_i) - x_i) = \sum_{i=1}^n \nabla_y \ell(y - x_i)\Big|_{y=f_{\Theta,c}(\tau_i)}.$$

c) KKT condition: In this section, we give the first-order optimal condition for our objective, which is a complex nonlinear equation system.

$$\begin{cases} P_{T_Q}(\nabla_Q F) = Q(\dfrac{Q^T\nabla_Q F - \nabla_Q^T F Q}{2}) + Q^{\perp}Q^{\perp\,T}\nabla_Q F = \mathbf{0}, \\ \nabla_\Theta F = \displaystyle\sum_{i=1}^n \mathrm{vech}(\tau_i\tau_i^T)\Big(\nabla_y \ell(y - x_i)^T\Big)\Big|_{y=f_{\Theta,c}(\tau_i)} V, \\ \nabla_c F = \displaystyle\sum_{i=1}^n \nabla_y \ell(y - x_i)\Big|_{y=f_{\Theta,c}(\tau_i)} = \mathbf{0}, \\ \nabla_{\tau_k} F = (U + V\Theta^T(M_{\tau_k} + N_{\tau_k}))^T \nabla_y \ell(y - x_k)|_{y=f(\tau_k)} = \mathbf{0}, \end{cases} \tag{12}$$

where $P_{T_Q}(\cdot)$ denotes the orthogonal projection onto the tangent space $T_Q\mathrm{St}(d+s, D)$ of the Stiefel manifold, ensuring that the stationarity condition with respect to Q is compatible with the orthogonality constraint.

Owing to the strong nonlinearity of the above system and the coupling among variables, directly solving the first-order conditions in (12) is intractable in practice. Instead, we adopt a gradient-based optimization strategy, where each variable is updated iteratively using (Riemannian) gradient descent, as detailed in Section VI.

IV. Convexity Analysis

We analyze the convexity of each subproblem with respect to the variables Θ, c, and $\{\tau_k\}$. When all other variables are fixed, the subproblem in c is convex, since the outer loss function $\ell(\cdot)$ is convex and the inner mapping is affine in c. The same argument applies to the subproblem in Θ. Consequently, the joint problem $\min_{\Theta,\, c}\ \sum_{i=1}^{n} \ell(f_{\Theta,c}(\tau_i) - x_i)$ is a convex optimization problem, as it consists of a sum of convex functions composed with affine mappings. As a result, for fixed $\{\tau_i\}$, any stationary point with respect to (Θ, c) is globally optimal.

In contrast, the subproblem with respect to τ_k is more challenging, as the inner transformation $f(\tau_k)$ is nonlinear. Nevertheless, we can still investigate the condition under which the Hessian is positive definite. Since the overall objective is separable across samples,

$$F(c, Q, \Theta, \Phi) = \sum_{k=1}^{n} \ell\big(f(\tau_k) - x_k\big),$$

the optimization with respect to $\{\tau_k\}_{i=1}^n$ decomposes into independent single-sample subproblems. Consequently, it suffices to analyze the convexity properties of the function $\tau \mapsto \ell\big(f(\tau) - x\big)$ for a single data point.

a) Convexity of the Projection Problem: Here, we investigate the optimization problem with respect to the lower-dimensional representation $\{\tau_k\}_{k=1}^n$. Since each subproblem with respect to τ_k is independent, for notational convenience, we leave out the index k and study on general problem as

$$\min_{\tau} \ell(f(\tau) - x).$$

We next derive the second-order derivative $H(\tau)$ with respect to the latent variable τ. Denote $y := f(\tau) - x$. The Hessian admits the decomposition as

$$H(\tau) = \nabla_\tau f(\tau)^\top \nabla_y^2 \ell(y) \nabla_\tau f(\tau) + \nabla_\tau^2 f(\tau) \times_1 \nabla_y \ell(y), \tag{13}$$

Here, $\nabla_y^2 \ell(y)$ denotes the Hessian of the loss with respect to y, and $\times_1$ represents the mode–1 tensor–vector contraction, i.e., the contraction of the first mode of the third-order tensor $\nabla_\tau^2 f(\tau)$ with the vector $\nabla_y \ell(y)$ evaluated at $y = f(\tau) - x$. For clarity, we summarize the dimensions of the involved quantities: $\nabla_\tau f(\tau) \in \mathbb{R}^{D\times d}, \nabla_\tau^2 f(\tau) \in \mathbb{R}^{D\times d\times d}, \nabla_y \ell(y)\big|_{y=f(\tau)-x} \in \mathbb{R}^D$.

The first term in (13) characterizes the curvature contribution induced by the loss function through the Jacobian of f, whereas the second term captures the intrinsic curvature of the model manifold arising from the nonlinear mapping f. Owing to the convexity of the loss function ℓ, the Hessian $\nabla_y^2 \ell(y)\big|_{y=f(\tau)-x}$ is positive semidefinite, which implies that the first term in (13) is also positive semidefinite. In particular, for the ℓ_p^p loss, the Hessian with respect to y admits the diagonal form

$$\nabla_y^2 \ell(y)\big|_{y=f(\tau)-x} = p(p-1)\,\mathrm{diag}\Big(\big\{|f(\tau)_i - x_i|^{p-2}\big\}_{i=1}^D\Big),$$

which is positive semidefinite for all $p \geq 1$.

As we restrict f from the quadratic form, the second-order derivative with respect to τ is constant and given by

$$\big\{\nabla_\tau^2 f(\tau)\big\}_{ruv} = 2\sum_s V_{rs}\, \mathcal{A}_{suv}.$$

where $\nabla_\tau^2 f(\tau) \in \mathbb{R}^{D\times d\times d}$. Therefore,

$$\begin{aligned} &\nabla_\tau^2 f(\tau) \times_1 \nabla_y \ell(y)\big|_{y=f(\tau)-x} \\ =&2\sum_{r=1}^{D}\{\sum_{t=1}^{s} V_{rt}\, \mathcal{A}_{tuv}\}\{\nabla_y \ell(y)\big|_{y=f(\tau)-x}\}_r. \end{aligned} \tag{14}$$

Owing to the presence of the curvature tensor $\mathcal{A}$ and its contraction with the gradient $\nabla_y \ell(y)$, the definiteness of the term $\nabla_\tau^2 f(\tau) \times_1 \nabla_y \ell(y)\big|_{y=f(\tau)-x}$ cannot be determined directly. Nevertheless, its magnitude can be controlled by analyzing the interaction between the curvature tensor $\mathcal{A}$ and the derivative of the loss function. In particular, for the ℓ_p^p loss with $p > 1$, either a sufficiently small gradient norm $\nabla_y \ell(y)|_{y=f(\tau)-x}$ (small noise setting) or a small curvature scale of the tensor $\mathcal{A}$ effectively suppresses the contribution of (14). As a consequence, the aggregated Hessian $H(\tau)$ remains positive definite over a relatively large region of the parameter space.

Motivated by this observation, we next establish a theorem that quantitatively characterizes the region in which $H(\tau)$ is positive definite by analyzing the structure of the Hessian matrix.

Theorem 1 (Local convexity radius for ℓ_p^p-SCQM) *Let $1 < p \leq 2$ and consider the objective*

$$\min_{\tau} \ell(\tau) = \|f(\tau) - x\|_p^p.$$

Assume there exist constants $\sigma_0, \rho, A_0 > 0$ such that, the following conditions hold:

1) *the Jacobian satisfies $\sigma_{\min}(\nabla_\tau f(\tau)) \geq \sigma_0$;*
2) *the residual is bounded away from zero, i.e., $\min_j |(f(\tau) - x)_j|^{p-2} \geq \rho$;*
3) *the quadratic tensor satisfies $\|\mathcal{A}\| := \max_s \|\mathcal{A}_s\| \leq A_0$.*

Then the Hessian of $\ell(\tau)$ is positive semidefinite throughout the $p-1$ norm ball $\mathcal{N}_{p-1}(x, r_p)$:

$$\mathcal{N}_{p-1}(x, r_p) \ :=\ \{\tau \in \mathcal{T} \mid \|f(\tau) - x\|_{p-1} \leq r_p\},$$

where $r_p := \Big(\frac{(p-1)\rho\sigma_0^2}{2A_0}\Big)^{\frac{1}{p-1}}$.

The proof for Theorem 1 is placed in the Appendix. It is worth noting that the assumption $\sigma_{\min}(\nabla_\tau f(\tau)) \geq \sigma_0$ is natural in our setting. Indeed, since $\nabla_\tau f(\tau) = U + V\Theta^\top(M_\tau + N_\tau)$, where U and V have orthonormal columns and satisfy $U^\top V = 0$, we have for any z,

$$\begin{aligned} \|\nabla_\tau f(\tau) z\|_2^2 =& \|Uz\|_2^2 + \|V\Theta^\top(M_\tau + N_\tau)z\|_2^2 \\ =& \|z\|_2^2 + \|\Theta^\top(M_\tau + N_\tau)z\|_2^2 \geq \|z\|_2^2. \end{aligned}$$

This immediately implies $\sigma_{\min}(\nabla_\tau f(\tau)) \geq 1$, and therefore one may take $\sigma_0 = 1$ without loss of generality. The assumption $\min_j |(f(\tau) - x)_j|^{p-2} \geq \rho$ is mild in practice when p is close to 2, as long as $(f(\tau) - x)_j \neq 0, \forall j$ and $\rho \in (0, 1)$ is chosen sufficiently small. In particular, in the

special case $p = 2$, the Hessian of the loss reduces to the identity matrix and the above condition holds trivially. Moreover, the boundedness assumption on the quadratic tensor, $\|\mathcal{A}\| := \max_s \|\mathcal{A}_s\| \leq A_0$, is natural, since A_0 can be taken as the maximum operator norm over all matrix slices $\{\mathcal{A}_s\}$ of the tensor $\mathcal{A}$.

V. Sensitivity analysis

Due to the high dimensionality of the second-order parameters and the presence of orthogonality constraints, directly conducting a full sensitivity analysis for all variables based on the KKT conditions in Eqn. (12) is analytically intractable. Instead, we adopt a simplified yet insightful approach by studying the sensitivity of the Fréchet mean [20] with respect to the input samples $\{x_i\}_{i=1}^n$. This setting can be regarded as a reduced version of our quadratic model, in which linear, curvature terms and manifold constraints are omitted, while the influence of the loss function remains explicit.

The Fréchet mean under a general loss function $\ell(\cdot)$ is defined in Eqn. (15). We view the optimizer c^* as an implicit function of the data $\{x_i\}_{i=1}^n$. By invoking the implicit function theorem [21], we characterize how c^* responds to small perturbations $\{\Delta x_i\}_{i=1}^n$ in the observations.

Proposition 2 (Sensitivity of the Optimal Solution) *Let $\ell : \mathbb{R}^D \to \mathbb{R}$ be a convex and twice continuously differentiable loss function, and let*

$$c^*(X) := \arg\min_{c\in\mathbb{R}^D} \sum_{i=1}^n \ell(x_i - c), \tag{15}$$

denote the optimal solution for the data matrix $X = \{x_i\}_{i=1}^n$. Define the residuals

$$r_i^* := x_i - c^*, \qquad i = 1, \ldots, n,$$

and the aggregated Hessian

$$H := \sum_{i=1}^n \nabla^2 \ell(r_i^*).$$

Then, the mapping $X \mapsto c^*(X)$ is locally Fréchet differentiable. In particular, for any perturbation $\{\Delta x_i\}_{i=1}^n$, the induced first-order variation of $c^*(X)$ is given by

$$\Delta c = H^{-1} \sum_{i=1}^n \nabla^2 \ell(r_i^*)\, \Delta x_i. \tag{16}$$

The proof for Proposition 2 is left in the appendix. Based on Proposition 2, we specialize the analysis to the ℓ_p^p loss with $1 < p \leq 2$ and to the ℓ_2 loss. This proposition allows us to elucidate why these loss functions yield estimators that are more robust than those based on the squared ℓ_2^2 loss. In particular, we show that the corresponding estimators possess bounded sensitivity to data perturbations, which significantly enhances their robustness to outliers.

A. Robustness Interpretation

Here, we explain why the ℓ_p^p loss for $1 < p \leq 2$ leads to enhanced robustness, based on the sensitivity result established in Proposition (2). For the ℓ_p^p loss, the Hessian $\nabla^2 \ell(r_i^*)$ is diagonal, with the k-th diagonal entry given by

$$\nabla^2 \ell(r_i^*) = p(p-1)\mathrm{diag}(\{|r_{ik}^*|^{p-2}\}_{k=1}^D).$$

For $p = 2$, the loss reduces to the squared Euclidean loss $\ell(r) = \|r\|_2^2$, and the Hessian is constant: $\nabla^2 \ell(r_i^*) = 2I_D$. Consequently, the aggregated Hessian satisfies $H = 2nI_D$, and the sensitivity formula in Proposition 2 simplifies to $\Delta c = -\frac{1}{n}\sum_{i=1}^n \Delta x_i$. This shows that the optimal intercept c^* responds linearly and uniformly to perturbations in the data, without any attenuation from the residual magnitudes.

For $1 < p \leq 2$, the Hessian weights depend on the residual magnitudes through $|r_{ik}^*|^{p-2}$. In particular, large residuals receive smaller weights, while small residuals are emphasized. As a result, the perturbation Δx_i is effectively reweighted in the sensitivity formula, leading to

$$\Delta c = H^{-1} \sum_{i=1}^n \mathrm{diag}(\{|r_{ik}^*|^{p-2}\}_{k=1}^D)\Delta x_i.$$

We observe that a perturbation Δx_{ik} in the k-th coordinate exerts a strong influence on the estimator when the corresponding residual $|r_{ik}^*|$ is small, while its influence becomes attenuated when $|r_{ik}^*|$ is large. This behavior arises because the sensitivity is weighted by the factor $|r_{ik}^*|^{p-2}$, whose exponent satisfies $p - 2 \leq 0$ for $1 < p \leq 2$. Consequently, large residuals receive diminishing weight in the sensitivity propagation, which explains the enhanced robustness of the ℓ_p^p loss compared to the squared Euclidean loss.

Next, we investigate the robustness properties of the ℓ_2 norm. For the ℓ_2 loss $\ell(r) = \|r\|_2$, when the residual $r \neq 0$, the Hessian admits the closed-form expression

$$\nabla^2 \ell(r) = \frac{1}{\|r\|_2}\left(I - \frac{rr^\top}{\|r\|_2^2}\right) = \frac{1}{\|r\|_2} P_{r^\perp},$$

where $P_{r^\perp}$ denotes the orthogonal projection matrix onto the subspace orthogonal to r. Substituting this expression into (16) yields the first-order sensitivity of the optimal intercept:

$$\Delta c = H^{-1} \sum_{i=1}^n \frac{1}{\|r_i^*\|_2} P_{r_i^{*\perp}}\, \Delta x_i. \tag{17}$$

This representation reveals two intrinsic robustness mechanisms of the ℓ_2 loss. First, perturbations associated with large residuals are attenuated by the factor $1/\|r_i^*\|_2$, thereby reducing the influence of outliers. Second, perturbations in the direction of the residual r_i^* do not affect Δc, since they are annihilated by the projection operator $P_{r_i^{*\perp}}$. Together, these properties explain the improved stability of ℓ_2-based estimators relative to the squared Euclidean loss.

VI. Algorithm

In this section, we present a gradient descent algorithm for solving the proposed generalized subspace-constrained quadratic model. The algorithm is highly flexible and accommodates a broad class of differentiable (or subdifferentiable)

loss functions; adapting it to a specific loss only requires replacing the gradient (or subgradient) term $\nabla_y \ell(y - x_i)$ accordingly.

Before introducing the algorithm, we describe two technical mechanisms that are incorporated to improve its convergence behavior. First, we employ an orthonormal retraction based on an aligned QR decomposition, which preserves the Stiefel manifold constraint and stabilizes the gradient flow. Second, we adopt an asynchronous learning-rate adjustment strategy, which enhances numerical stability and accelerates the optimization process.

a) Orthonormal retraction via aligned QR: Due to the orthogonality constraint imposed on Q, the Euclidean gradient update reduces to an update along the Riemannian gradient direction G_Q, which is obtained by projecting the Euclidean gradient onto the tangent space at Q. To maintain the orthonormality constraint, the updated matrix is then retracted back onto the Stiefel manifold via a QR decomposition [19]. However, the QR decomposition is not unique: for a given matrix, there may exist two orthonormal matrices Q and Q', together with two upper triangular matrices R and R', such that $QR = Q'R'$. To remove this ambiguity, we impose a standard sign convention on the diagonal entries of the upper triangular factor, requiring them to be positive, as discussed in [22]. Under this convention, the QR decomposition is unique, and the associated retraction map is continuous. Consequently, the update sequence $\{Q^{(t)}\}$ satisfies

$$\lim_{\eta_t \to 0} \left\| Q^{(t+1)} - Q^{(t)} \right\|_{\mathrm{F}} = 0,$$

ensuring that successive iterates vary smoothly as the step size diminishes. This continuity property is essential for guaranteeing a monotonic decrease of the objective function during the gradient descent process and for maintaining the stability of the optimization on the Stiefel manifold.

b) Asynchronous learning-rate adjustment: An appropriate choice of the learning rate is crucial for both convergence behavior and computational efficiency. However, the optimal learning rate is influenced by multiple factors, including the local geometric properties of the objective function at the current iterate and the curvature induced by the constraint manifold. Asynchronous learning-rate adjustment enables the use of distinct learning rates for different parameter blocks—such as η_Q for the subspace variable Q and η_Θ for the quadratic parameters Θ—allowing each component to be updated in accordance with its own geometric and optimization characteristics.

To ensure an effective learning rate, we adopt an asynchronous learning-rate adjustment strategy. Specifically, we initialize separate learning rates $\eta_c, \eta_Q, \eta_\Theta$, and η_τ for the variables c, Q, Θ, and τ_i, respectively. During each update, the learning rate associated with a given variable is adjusted independently based on a sufficient decrease criterion.

We employ a backtracking line search strategy based on the Armijo rule to adaptively determine suitable step sizes during optimization [23]. Specifically, when updating the variable c, we first take a tentative gradient step and evaluate the objective function $F(c - \eta_c \nabla_c F,\ \Theta,\ Q,\ \{\tau_i\})$. This value is compared against the sufficient decrease condition $F(c, \Theta, Q, \{\tau_i\}) - \alpha\eta_c \left\| \nabla_c F(c, \Theta, Q, \{\tau_i\}) \right\|_{\mathrm{F}}^2$, where $\alpha \in (0, 1)$ is a prescribed constant. If the condition is satisfied, the step size η_c is accepted; otherwise, it is reduced, for example by setting $\eta_c \leftarrow \eta_c/2$, and the test is repeated. The same backtracking procedure is applied independently to the updates of Q, Θ, and $\{\tau_i\}$. This block-wise adaptive step-size strategy allows each parameter group to adjust to the local geometry of the objective function and the associated constraint manifold,

Algorithm 1 Riemannian Gradient Descent for SCQM

Require: $X = [x_1, \ldots, x_n] \in \mathbb{R}^{D \times n}$; max iterations T; tolerance ε; initial step sizes $(\eta_Q, \eta_\Theta, \eta_c, \eta_\tau)$; loss ℓ with (sub)gradient $\partial\ell(\cdot)$; retraction $\mathrm{Retr}_{\mathrm{St}}(\cdot)$ and $\mathrm{sym}(A) = \frac{1}{2}(A + A^T)$.

Ensure: $Q = [U, V] \in \mathrm{St}(d+s, D)$, Θ, c, $\{\tau_i\}_{i=1}^n$.

Initialize: $c^{(0)} = \frac{1}{n} X \mathbf{1}_n$; compute PCA of $X - c^{(0)}\mathbf{1}_n^T$ and set $Q^{(0)} = [U^{(0)}, V^{(0)}] \in \mathrm{St}(d+s, D)$; $\tau_i^{(0)} = (U^{(0)})^T (x_i - c^{(0)})$; $\Theta^{(0)} \sim \mathcal{N}(0, \sigma^2 I)$.

for $t = 0, \ldots, T-1$ **do**

Residuals and weights:

for $i = 1, \ldots, n$

$$\phi_i^{(t)} = \mathrm{vech}\big(\tau_i^{(t)} \tau_i^{(t)T}\big),$$
$$f_i^{(t)} = c^{(t)} + U^{(t)} \tau_i^{(t)} + V^{(t)} \Theta^{(t)T} \phi_i^{(t)},$$
$$r_i^{(t)} = f_i^{(t)} - x_i, \qquad g_i^{(t)} \in \partial\ell\big(r_i^{(t)}\big).$$

end for

Euclidean gradients:

$$\nabla_c F^{(t)} = \sum_{i=1}^n g_i^{(t)},$$
$$\nabla_\Theta F^{(t)} = \sum_{i=1}^n \phi_i^{(t)} \big(V^{(t)T} g_i^{(t)}\big)^T,$$
$$\nabla_Q F^{(t)} = \sum_{i=1}^n g_i^{(t)} \begin{bmatrix} \tau_i^{(t)T} & \phi_i^{(t)T} \Theta^{(t)} \end{bmatrix}.$$

Riemannian step on $\mathrm{St}(d+s, D)$**:**

$$G_Q^{(t)} = \nabla_Q F^{(t)} - Q^{(t)} \mathrm{sym}\big(Q^{(t)T} \nabla_Q F^{(t)}\big),$$
$$Q^{(t+1)} = \mathrm{Retr}_{\mathrm{St}}\big(Q^{(t)} - \eta_Q^{(t)} G_Q^{(t)}\big),$$

extract $Q^{(t+1)} = [U^{(t+1)}, V^{(t+1)}]$.

Euclidean updates:

$$\Theta^{(t+1)} = \Theta^{(t)} - \eta_\Theta^{(t)} \nabla_\Theta F^{(t)},$$
$$c^{(t+1)} = c^{(t)} - \eta_c^{(t)} \nabla_c F^{(t)}.$$

Update latent coordinates:

for $i = 1, \ldots, n$

$$J_{\tau_i}^{(t)} = \left. \frac{\partial f_i^{(t)}}{\partial \tau_i} \right|_{(U^{(t+1)}, V^{(t+1)}, \Theta^{(t+1)}, \tau_i^{(t)})},$$
$$\tau_i^{(t+1)} = \tau_i^{(t)} - \eta_\tau^{(t)}\, J_{\tau_i}^{(t)T} g_i^{(t)}.$$

end for

end for

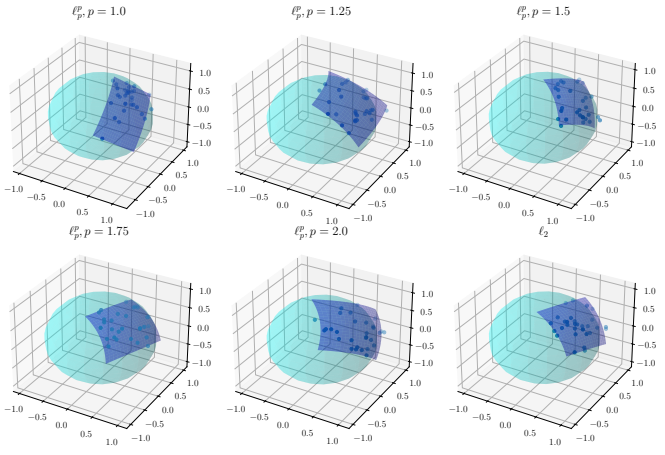

Fig. 3. Illustration of the fitted curves and projection points obtained using ℓ_p^p losses with different values of p.

thereby improving numerical stability and accelerating the convergence of the overall optimization algorithm.

c) Learning process: The learning process can be summarized as: Given a data matrix $X \in \mathbb{R}^{D\times n}$, the algorithm jointly estimates the latent representations $\{\tau_i\}_{i=1}^n$, the quadratic mapping parameters Θ, the mean vector c, and an orthonormal basis $Q \in \mathrm{St}(d+s, D)$.

The algorithm initializes c as the column mean of X, and initializes Q using the leading $d+s$ left singular vectors of the centered data matrix $X - c\mathbf{1}_n^T$. Each latent variable τ_i is initialized by projecting the centered data point $x_i - c$ onto the first d columns of Q, while Θ is initialized randomly.

At each iteration, the Euclidean gradients with respect to all variables—$\nabla_Q F, \nabla_\Theta F, \nabla_c F$, and $\{\nabla_{\tau_i} F\}_{i=1}^n$—are first computed. To enforce the orthonormality constraint on Q, the gradient $\nabla_Q F$ is projected onto the tangent space of the Stiefel manifold, yielding the Riemannian gradient direction. A Riemannian gradient step is then performed, followed by a retraction onto the Stiefel manifold via QR decomposition. A backtracking line search is applied along the retracted Riemannian gradient direction to determine a suitable step size. For the unconstrained parameters Θ and c, standard Euclidean gradient updates are employed.

For the update of each τ_i, we also employ an asynchronous backtracking line search to determine an acceptable step size that ensures sufficient decrease in the objective value. In other words, the step size associated with each τ_i is selected independently. Each latent variable τ_i is then updated via a gradient step that explicitly incorporates the structured matrices M_{τ_i} and N_{τ_i}, which arise from the quadratic term of the model. The procedure is iterated until convergence or until a prescribed maximum number of iterations is reached, yielding a structured quadratic approximation that minimizes the chosen loss function.

VII. Numerical Experiments

In this section, we demonstrate how the proposed model can be applied to the task of manifold denoising, or more generally, manifold data refinement. Unlike the toy example in Section II-E, where different noise models are paired with their corresponding loss functions, here we evaluate all methods under the same noise distribution. Specifically, we assess the performance of various methods under additive Gaussian noise with different noise scales.

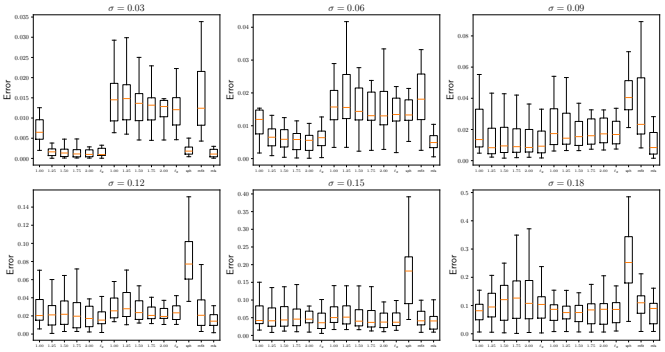

Fig. 4. Performance is compared across models and noise levels using boxplots of the squared reconstruction error, $\|\tilde{x}_i - P(x_i)\|_2^2$, computed over all samples.

In this experimental setting, we assume that the observed dataset $\{x_i\}$ is generated by corrupting a noise-free dataset $\{\tilde{x}_i\}$ with Gaussian noise $\{\epsilon_i\}$, such that

$$x_i = \tilde{x}_i + \epsilon_i, \quad \text{where} \quad \tilde{x}_i \sim \mathcal{M},\ \epsilon_i \sim \mathcal{N}(\mathbf{0}, \sigma^2 I). \tag{18}$$

We implement data refinement in two steps. First, for each observation x_i, we identify its K nearest neighbors, denoted as $\mathcal{N}_{x_i}$. Second, we learn a subspace-constrained quadratic matrix factorization model to find a local representation $\widehat{f}$, which is parameterized by $\widehat{Q}, \widehat{\Theta}, \widehat{c}$. Next, we project x_i back onto the learned subspace $\widehat{f}$ by solving the following optimization problem:

$$P(x_i) = \arg \min_{x \in \{\widehat{f}(\tau), \tau \in \mathbb{R}^d\}} \|x_i - x\|_p^p.$$

Finally, we evaluate the performance of the manifold approximation by measuring the empirical mean squared error:

$$\mathcal{E} = \frac{1}{N} \sum_{i=1}^{N} \|\tilde{x}_i - P(x_i)\|_2^2.$$

A smaller value of $\mathcal{E}$ indicates a stronger recovery capability of the manifold approximation method.

All code and implementations are publicly available in our GitHub repository[1].

A. Simulation with Spherical Data in $\mathbb{R}^3$

Here, we compare the performance of the SCQM model with that of its competitors, including SPH, MFIT, and MLS, under different loss functions and across varying noise levels. We uniformly sample 300 points on the 3D sphere as the ground-truth signals $\tilde{x}_i$. Noisy observations are then generated according to (18). We evaluate model performance under multiple noise settings by varying the noise standard deviation $\sigma \in \{0.05, 0.10, 0.15, 0.20\}$. For each noise level, we test ℓ_p^p

[1]https://github.com/zhaizheng/Robust_SCQM.git

TABLE II
MEAN SQUARED ERROR (MSE) OF DIFFERENT MODELS UNDER VARYING NOISE LEVELS.

	Quadratic model: $\Theta \neq \mathbf{0}$						Linear model: $\Theta = \mathbf{0}$						Other methods		
	ℓ_p^p					ℓ_2	ℓ_p^p					ℓ_2	SPH	MFIT	MLS
σ \ p	1.00	1.25	1.50	1.75	2.00	2.00	1.00	1.25	1.50	1.75	2.00	2.00	–	–	–
0.03	0.009	0.002	0.002	0.001	0.001	0.002	0.016	0.016	0.014	0.013	0.013	0.013	0.002	0.018	0.001
0.06	0.014	0.006	0.006	0.005	0.005	0.006	0.021	0.020	0.017	0.015	0.015	0.014	0.018	0.022	0.005
0.09	0.020	0.016	0.017	0.016	0.016	0.015	0.023	0.022	0.021	0.021	0.021	0.021	0.044	0.038	0.014
0.12	0.030	0.023	0.025	0.022	0.021	0.020	0.033	0.035	0.027	0.025	0.024	0.026	0.084	0.028	0.016
0.15	0.070	0.061	0.064	0.067	0.071	0.053	0.072	0.074	0.061	0.056	0.055	0.057	0.180	0.061	0.053
0.18	0.083	0.100	0.144	0.132	0.137	0.099	0.087	0.085	0.082	0.084	0.087	0.090	0.262	0.113	0.085
0.21	0.126	0.139	0.232	0.216	0.204	0.140	0.129	0.140	0.133	0.125	0.126	0.132	0.317	0.140	0.122

loss functions with $p \in \{1.0, 1.25, 1.50, 1.75, 2.0\}$, as well as the ℓ_2 loss. For fairness and completeness, all methods are implemented within a local neighborhood of each sample x_i, where the neighborhood contains exactly 30 neighboring samples. The per-sample squared error $\|\tilde{x}_i - P(x_i)\|_2^2$ is summarized using boxplots for each combination of model and noise level, as shown in Fig. 4 and Fig. 3.

From the observations in Table II and Figure 4, we can conclude that :

- When an appropriate value of p is selected, our method consistently outperforms state-of-the-art approaches, including SPH, MFIT, and MLS, across a wide range of noise levels.
- In the low-noise regime (e.g., $\sigma < 0.15$), the quadratic model consistently outperforms the corresponding linear model, which in turn validates the effectiveness of incorporating quadratic terms. However, as the noise level increases, the performance of the quadratic model deteriorates. For instance, at $\sigma = 0.18$, the quadratic model underperforms the linear model, likely due to overfitting, as it captures spurious patterns arising from the noise rather than the underlying signal.
- For the quadratic model under the ℓ_p^p loss, larger values of p are favorable in the low-noise regime (e.g., $\sigma < 0.12$). However, as the noise level further increases, smaller values of p become more advantageous. This observation highlights a fundamental trade-off between accurate signal projection and robustness to noise. Although minimizing the ℓ_p^p loss with $p = 1$ does not recover the true signal under the minimum-distance projection criterion, it exhibits the strongest robustness among the considered methods.
- The quadratic model with the ℓ_2 loss exhibits stable and well-balanced performance across different noise levels, demonstrating its ability to effectively handle noise and recover the true projection in Euclidean space, as it avoids the additional sensitivity introduced by higher-order power operations.
- When the noise level is small, all methods perform well. However, the ℓ_p^p loss with $p = 1$ consistently underperforms relative to the other metrics, particularly at $\sigma = 0.05$ and $\sigma = 0.10$. This is likely due to a mismatch between the noise distribution and the ℓ_1 metric; in other words, an ℓ_1-based projection does not coincide with the true projection under the Euclidean distance. As the noise level increases, the ℓ_2 model and the ℓ_p^p models with smaller values of p tend to outperform the alternatives. This behavior can be attributed to improved robustness: these losses are less sensitive to large-variance noise and outlier-contaminated samples, leading to more stable performance under heavier noise.

Fig. 5. Comparison of reconstruction methods under different loss functions and model settings. Each row displays 16 reconstructed images. From top to bottom, the rows correspond to the original images, linear model with ℓ_2 loss, SCQM with ℓ_2 loss, linear model with ℓ_p^p loss ($p = 2$), SCQM with ℓ_p^p loss ($p = 2$), linear model with ℓ_p^p loss ($p = 1$), and SCQM with ℓ_p^p loss ($p = 1$), respectively.

B. Performance on Real World Dataset

Here, we demonstrate the performance of the robust SCQM on a real-world dataset using handwritten MNIST [24] images. We randomly select 100 samples consisting of two easily confused digits, namely '4' and '9'. The original images are projected onto a low-dimensional manifold with latent dimension $d = 2$. We visualize the reconstructed images corresponding to these projections and examine the reconstruction details across different model settings.

We evaluate three SCQM models with different loss functions, namely ℓ_2 (third row), ℓ_2^2 (fifth row), and ℓ_1 (seventh row), as shown in Fig. 5. In addition, we evaluate the corresponding linear variants by setting $\Theta = \mathbf{0}$ throughout the learning process, thereby removing the quadratic component. These linear counterparts are displayed in the second, fourth, and sixth rows, corresponding to the three loss functions, respectively. This comparison serves as an ablation study to systematically assess the effectiveness and contribution of the quadratic term within the SCQM framework.

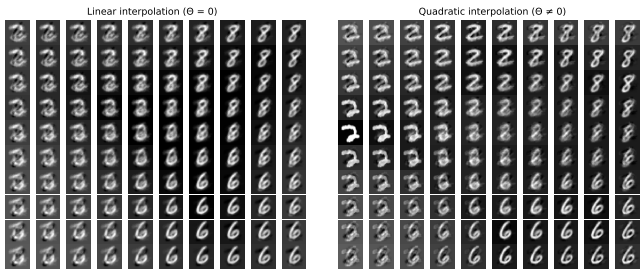

Fig. 6. Visualization of latent-space ($d = 2$) interpolation for the linear model ($\Theta = \mathbf{0}$) and the quadratic model ($\Theta \neq \mathbf{0}$), learned from data consisting of the three digits '2', '6', and '8'.

From Fig. 5, we draw the following conclusions. First, by comparing the linear and quadratic variants—specifically, the second versus third rows, the fourth versus fifth rows, and the sixth versus seventh rows—we observe that the inclusion of the quadratic term significantly improves the discrimination between the digits '4' and '9' with the improvement being particularly evident in the second column from the right. Second, the ℓ_1 loss and the ℓ_2 loss consistently outperform the ℓ_2^2 loss (SQMF discussed in [6]), producing sharper and more visually coherent reconstructions. This highlights their superior robustness and reconstruction capability in the presence of ambiguous or overlapping digit structures.

C. Interpolation via SCQM

Finally, we demonstrate that SCQM can also be used as an interpolation model for generating new high-dimensional data. Since we have learned a mapping $\widehat{f} : \mathbb{R}^d \to \mathbb{R}^D$, uniformly sampling points in the latent space $\mathbb{R}^d$ allows us to visualize the global landscape of $\widehat{f}$ through its outputs in the ambient space. This interpolation procedure further serves as a means to evaluate the expressive capability of different models, as it reveals how well each learned mapping captures the underlying manifold structure.

To validate the strong expressive capability of the proposed SCQM in comparison with its linear counterpart in the latent space. We randomly select images corresponding to the digits '2', '6', and '8', and learn an SCQM model $\widehat{f}$ with latent dimension $d = 2$ and quadratic dimension $s = 20$. We then perform interpolation in the latent space by uniformly sampling between the minimum and maximum values along each latent dimension, resulting in a grid of latent points $\{\tau_k\}$. Each interpolation point is mapped back to the image space via the learned decoder $\widehat{f} : \tau \mapsto I$, and the reconstructed images are visualized at their corresponding latent locations. The results are shown in Fig. 6. As illustrated, the quadratic component plays a crucial role in modeling nonlinear variations of the data manifold. While the linear model produces linear transitions, the quadratic model yields smoother and more continuous interpolations, better capturing the intrinsic geometric structure of the data.

VIII. Conclusion

This work investigates a robust quadratic fitting framework that generalizes subspace-constrained quadratic factorization. This generalization provides a promising approach for addressing scenarios with a limited number of observed data points and for relaxing the local flatness assumption inherent in linear subspace models. We propose a gradient descent method to solve the resulting robust quadratic fitting problem and conduct a sensitivity analysis for the Fréchet mean problem, which can be viewed as a special case within the quadratic function class. Numerical experiments are conducted to validate the effectiveness of the proposed framework.

Several directions remain open for future investigation. First, the statistical properties of the proposed quadratic model have not been analyzed. Future work could establish non-asymptotic results [25] with respect to the sample size n, intrinsic dimension d, and additional parameters such as the loss exponent p. Second, although the gradient descent algorithm exhibits stable behavior in our experiments, its theoretical convergence properties have not been studied. Providing convergence guarantees under appropriate conditions is an important direction for future research. Third, our current analysis focuses on assessing the benefit of incorporating a curvature (quadratic) term relative to the linear case. Extending the framework to more expressive models and exploring richer nonlinear structures may further improve performance and is another promising avenue for future work.

References

[1] Nanda Kambhatla and Todd Leen. Dimension reduction by local principal component analysis. *Neural Computation*, 9:1493–1516, 10 1997.

[2] Christopher R Genovese, Marco Perone-Pacifico, Isabella Verdinelli, Larry Wasserman, et al. Nonparametric ridge estimation. *The Annals of Statistics*, 42(4):1511–1545, 2014.

[3] Charles Fefferman, Sergei Ivanov, Yaroslav Kurylev, Matti Lassas, and Hariharan Narayanan. Fitting a putative manifold to noisy data. In *Conference On Learning Theory*, pages 688–720, 2018.

[4] Umut Ozertem and Deniz Erdogmus. Locally defined principal curves and surfaces. *Journal of Machine learning research*, 12(Apr):1249–1286, 2011.

[5] Zheng Zhai, Hengchao Chen, and Qiang Sun. Quadratic matrix factorization with applications to manifold learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(9):6384–6401, 2024.

[6] Zheng Zhai and Xiaohui Li. Subspace-constrained quadratic matrix factorization: Algorithm and applications. *Pattern Recognition*, 161:111333, 2025.

[7] Guangcan Liu, Zhouchen Lin, and Yong Yu. Robust subspace segmentation by low-rank representation. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 663–670, 2010.

[8] Yongqiang Zhang, Daming Shi, Junbin Gao, and Dansong Cheng. Low-rank-sparse subspace representation for robust regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7445–7454, 2017.

[9] Christopher R. Genovese, Marco Perone-Pacifico, Isabella Verdinelli, and Larry Wasserman. Nonparametric ridge estimation. *Annals of Statistics*, 42(4):1511–1545, 2014.

[10] Umut Ozertem and Deniz Erdogmus. Locally defined principal curves and surfaces. *Journal of Machine Learning Research*, 12:1249–1286, 2011.

[11] Charles Fefferman, Sergei Ivanov, Yaroslav Kurylev, Matti Lassas, and Hariharan Narayanan. Fitting a putative manifold to noisy data. In *Conference on Learning Theory*, pages 688–720, 2018.

[12] Barak Sober and David Levin. Manifold approximation by moving least-squares projection. *Constructive Approximation*, 52(3):433–478, 2020.

[13] Hengchao Chen and Zheng Zhai. Power transformed density ridge estimation. *IEEE Signal Processing Letters*, 2025.

[14] Alexey A Roenko, Vladimir V Lukin, I Djurović, and M Simeunović. Estimation of parameters for generalized gaussian distribution. In *2014 6th International Symposium on Communications, Control and Signal Processing (ISCCSP)*, pages 376–379. IEEE, 2014.

[15] Frédéric Pascal, Lionel Bombrun, Jean-Yves Tourneret, and Yannick Berthoumieu. Parameter estimation for multivariate generalized gaussian distributions. *IEEE Transactions on Signal Processing*, 61(23):5960–5971, 2013.
[16] Minh N Do and Martin Vetterli. Wavelet-based texture retrieval using generalized gaussian density and kullback-leibler distance. *IEEE transactions on image processing*, 11(2):146–158, 2002.
[17] Samuel Kotz, Tomasz Kozubowski, and Krzystof Podgorski. *The Laplace distribution and generalizations: a revisit with applications to communications, economics, engineering, and finance*. Springer Science & Business Media, 2012.
[18] Gilbert Strang. *Introduction to linear algebra*. SIAM, 2022.
[19] P-A Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2008.
[20] Abhishek Bhattacharya and Rabi Bhattacharya. *Nonparametric inference on manifolds: with applications to shape spaces*, volume 2. Cambridge University Press, 2012.
[21] Steven G. Krantz and Harold R. Parks. *The Implicit Function Theorem: History, Theory, and Applications*. Birkhäuser, 2013.
[22] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 4th edition, 2013.
[23] Jorge Nocedal. Numerical optimization. *Springer Ser. Oper. Res. Financ. Eng./Springer*, 2006.
[24] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 2002.
[25] Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge university press, 2019.

APPENDIX

Proof (Proposition 1) *Starting from the first-order optimality condition* (11)*, we left-multiply both sides by* $Q^\top$ *to obtain*

$$Q^\top \nabla_Q F(c, Q, \Theta, \Phi) - 2Q^\top Q\Lambda = \mathbf{0}.$$

Using the orthonormality constraint $Q^\top Q = I_{d+s}$*, this yields*

$$\Lambda = \tfrac{1}{2} Q^\top \nabla_Q F(c, Q, \Theta, \Phi).$$

Substituting this expression for Λ *back into* (11)*, we obtain*

$$\nabla_Q F(c, Q, \Theta, \Phi) - QQ^\top \nabla_Q F(c, Q, \Theta, \Phi) = \mathbf{0}. \tag{19}$$

Let $Q^\perp$ *denote an orthonormal complement of* Q *such that*

$$QQ^\top + Q^\perp (Q^\perp)^\top = I.$$

Decomposing the Euclidean gradient as

$$\nabla_Q F = QQ^\top \nabla_Q F + Q^\perp (Q^\perp)^\top \nabla_Q F,$$

and substituting into (19)*, we obtain*

$$Q\big(Q^\top \nabla_Q F - \nabla_Q^\top F Q\big) + Q^\perp (Q^\perp)^\top \nabla_Q F = \mathbf{0}.$$

Since the column spaces of Q *and* $Q^\perp$ *are orthogonal, the above equality implies*

$$Q^\top \nabla_Q F - \nabla_Q^\top F Q = \mathbf{0}, \qquad (Q^\perp)^\top \nabla_Q F = \mathbf{0}.$$

These conditions are exactly equivalent to the vanishing of the Riemannian gradient on the Stiefel manifold, i.e., $G_Q = \mathbf{0}$*.*

Conversely, if the Riemannian gradient vanishes, then the Euclidean gradient satisfies

$$\nabla_Q F - Q\,\mathrm{sym}(Q^\top \nabla_Q F) = \mathbf{0}.$$

Setting $\Lambda = \frac{1}{2} Q^\top \nabla_Q F$*, which is symmetric, it is straightforward to verify that the first-order condition* (11) *holds.*

Proof (Theorem 1) *From* (13)*, the Hessian with respect to* τ_k *admits the decomposition* $H(\tau) = H_1(\tau) + H_2(\tau)$*, where*

$$H_1(\tau) = \nabla_\tau f(\tau)^\top \nabla_r^2 \ell_p^p(r)\big|_{r=f(\tau)-x} \nabla_\tau f(\tau),$$
$$H_2(\tau) = \nabla_\tau^2 f(\tau) \times_1 \nabla_r \ell_p^p(r)\big|_{r=f(\tau)-x}.$$

First, we give a lower bound for $H_1(\tau)$*. For the* ℓ_p^p *loss with* $1 < p < 2$*, the Hessian with respect to* y *satisfies*

$$\nabla_r^2 \ell_p^p(r) = p(p-1)\,\mathrm{diag}(|r_1|^{p-2}, ..., |r_D|^{p-2}) \succeq p(p-1)\rho I_D,$$

Using Assumptions (1) and (2), we obtain the uniform lower bound

$$\lambda_{\min}\big(H_1(\tau)\big) \geq p(p-1)\rho\sigma_0^2.$$

For $H_2(\tau)$*, since* V *has orthonormal columns and the tensor–vector contraction satisfies* $\|\mathcal{A} \times_1 u\|_{\mathrm{op}} \leq \|\mathcal{A}\|\,\|u\|_1$*, we have* $\|V\mathcal{A}\| = \|\mathcal{A}\| \leq A_0$*. Therefore,*

$$\|H_2(\tau)\| \leq 2pA_0 \sum_{d=1}^{D} |(f(\tau)-x)_d|^{p-1} = 2pA_0 \|f(\tau)-x\|_{p-1}^{p-1}.$$

By Weyl's inequality, $\lambda_{\min}\big(H(\tau)\big) \geq \lambda_{\min}\big(H_1(\tau_k)\big) - \|H_2(\tau)\|$*. Hence,* $H(\tau) \succeq 0$ *when*

$$p(p-1)\rho\sigma_0^2 \geq 2pA_0 \|f(\tau) - x\|_{p-1}^{p-1}.$$

This condition holds for all $\|f(\tau) - x\|_{p-1} \leq r_p$*, where* $r_p = \left(\frac{(p-1)\rho\sigma_0^2}{2A_0}\right)^{\frac{1}{p-1}}$ *..*

Proof (Proposition 2) *We begin with the first-order optimality condition satisfied by the minimizer* $c^*(X)$*:*

$$\nabla_c F(c^*, X) = -\sum_{i=1}^{n} \nabla \ell\big(x_i - c^*\big) = \mathbf{0}. \tag{20}$$

Define

$$\Psi(c, X) := \nabla_c F(c, X).$$

Since ℓ *is twice continuously differentiable,* Ψ *is continuously differentiable in a neighborhood of* (c^*, X)*. Moreover, under the stated assumptions, the Jacobian* $\nabla_c \Psi(c^*, X)$ *is nonsingular. By the implicit function theorem, the equation* $\Psi(c, X) = \mathbf{0}$ *implicitly defines* c *as a differentiable function of the data matrix* X *in a neighborhood of* X*, which we denote by* $c(X)$*.*

Differentiating the identity $\Psi(c(X), X) = \mathbf{0}$ *with respect to* X *yields*

$$\nabla_c \Psi\big(c(X), X\big)\, \nabla_X c(X) + \nabla_X \Psi\big(c(X), X\big) = \mathbf{0}.$$

Solving for $\nabla_X c(X)$ *gives*

$$\nabla_X c(X) = -\big(\nabla_c \Psi\big(c(X), X\big)\big)^{-1} \nabla_X \Psi\big(c(X), X\big). \tag{21}$$

We now compute the required Jacobians explicitly. Differentiating Ψ *with respect to* c *yields*

$$\nabla_c \Psi(c^*, X) = \sum_{i=1}^{n} \nabla^2 \ell(r_i^*), \qquad r_i^* := x_i - c^*. \tag{22}$$

By convexity of ℓ*, this matrix is positive definite and hence invertible. Next, consider a perturbation* $\Delta X = \{\Delta x_i\}_{i=1}^n$*.*

The directional derivative of Ψ with respect to X in the direction ΔX is given by

$$D_X \Psi(c^*, X)[\Delta X] = -\sum_{i=1}^{n} \nabla^2 \ell(r_i^*)\, \Delta x_i. \tag{23}$$

Substituting (22) *and* (23) *into* (21)*, we obtain*

$$\Delta c = \left(\sum_{i=1}^{n} \nabla^2 \ell(r_i^*) \right)^{-1} \sum_{i=1}^{n} \nabla^2 \ell(r_i^*)\, \Delta x_i,$$

which establishes the desired sensitivity formula and completes the proof.