On the Curvature of Pattern Transformation Manifolds: Numerical Estimation and Applications

E. Kokiopoulou and **D. Kressner**
Seminar for Applied Mathematics
Department of Mathematics
ETH Zurich, CH-8092, Zurich
{effrosyni.kokiopoulou,daniel.kressner}@math.ethz.ch.

P. Frossard
Signal Processing Laboratory (LTS4)
Institute of Electrical Engineering
EPFL, CH-1015, Lausanne
pascal.frossard@epfl.ch

Abstract

This paper addresses the numerical estimation of the principal curvature of pattern transformation manifolds. When a visual pattern undergoes a geometric transformation, it forms a (sub)manifold in the ambient space, which is usually called the transformation manifold. The manifold curvature is an important property characterizing the manifold geometry, with several applications in manifold learning. We propose an efficient numerical algorithm for estimating the principal curvature at a certain point on the transformation manifold.

Introduction

We study the problem of estimating numerically the principal curvature of pattern transformation manifolds, which are formed by patterns that undergo geometric transformations (e.g., rotations, scaling and so on). The principal curvature is defined as the largest eigenvalue of an appropriately defined linear operator (see below) and has recently seen several uses in manifold learning. For example, the curvature can be a valuable tool for manifold discretization towards transformation invariant pattern recognition. In particular, the works (Pozdnoukhov and Bengio; Kokiopoulou, Pirillos, and Frossard) introduce geometrically transformed versions of the data samples, known as *virtual samples*, to make graph-based classification methods robust to geometric transformations. However, one faces in this case the problem of constructing the virtual samples or equivalently, the problem of manifold discretization.

In the context of compressed sensing, it has been shown in (Baraniuk and Wakin) that the manifold condition number, which is closely related to the curvature, is an important factor towards characterizing the number of measurements that are needed for obtaining an isometric embedding of the manifold under random projections. Therefore, the manifold curvature has received increasing attention in manifold learning. We propose below an efficient and simple numerical algorithm for computing the principal curvature at a certain point on the transformation manifold.

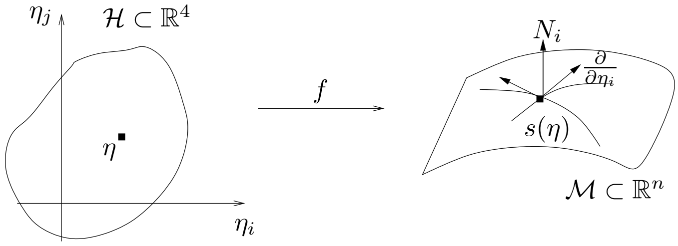

Figure 1: The parameter space $\mathcal{H}$ provides a parametrization of the transformation manifold $\mathcal{M}$.

Numerical estimation of the curvature

A *transformation manifold* $\mathcal{M}$ consists of all geometric transformations of a certain image $s \in \mathbb{R}^n$. Letting $\eta \in \mathcal{H}$ denote the parameters describing such a geometric transformation, this manifold can be expressed mathematically as

$$\mathcal{M} = \{s(\eta) := U(\eta)s,\ \eta \in \mathcal{H}\}, \qquad (1)$$

where $U(\eta)$ is an operator that acts on s and maps it to the transformed pattern $s(\eta)$. Although $\mathcal{M}$ resides in a high-dimensional space, its intrinsic dimension d is rather small and equal to the number of transformation parameters.

In the following, we provide the necessary prerequisites for defining and analyzing the principal curvature of parametric manifolds $\mathcal{M}$ of the form (1). The metric tensor $G \in \mathbb{R}^{d\times d}$ is given by $[G]_{ij} = \langle t_i, t_j \rangle$, where t_i, t_j are the ith and jth tangent vectors, defined as $t_i = \frac{\partial s(\eta)}{\partial \eta_i}$ and assumed to be linearly independent. The tangent space $T_\eta\mathcal{M}$ at point $s(\eta) \in \mathcal{M}$ is given by $T_\eta\mathcal{M} = \operatorname{span}\{t_1, \dots, t_d\}$. Since $d = \dim T_\eta\mathcal{M}$ and $\mathcal{M} \subset \mathbb{R}^n$, the codimension of $T_\eta\mathcal{M}$ is $n-d$. Consider the direct sum $\mathbb{R}^n = T_\eta\mathcal{M} \oplus T_\eta\mathcal{M}^\perp$ and let $\{N_1, \dots, N_{n-d}\}$ be an orthonormal basis of $T_\eta\mathcal{M}^\perp$. Then any (unit) normal vector can be written as $N = \sum_{i=1}^{n-d} \zeta^i N_i$, with coefficients $\zeta^i = \langle N_i, N \rangle$ (see Fig. 1 for a graphical illustration).

In order to define the principal curvature, we need to define first the linear operator $L_\zeta : T_\eta\mathcal{M} \to T_\eta\mathcal{M}$ associated

Algorithm 1 Numerical estimation of the curvature

1: **Input**: transformed pattern $s(\eta)$, normal coordinates $\zeta^1, \ldots, \zeta^{n-d}$
2: **Output**: estimate λ_ζ of the principal curvature.
3: Compute $t_i = \frac{\partial s}{\partial \eta_i}$, $i = 1, \ldots, d$.
4: Compute $t_{ij} = \frac{\partial^2 s}{\partial \eta_i \partial \eta_j}$, $i, j = 1, \ldots, d$.
5: Compute $\left[G\right]_{ij} = \langle t_i, t_j \rangle$.
6: Compute the entries g^{ij} of G^{-1}.
7: Build an orthonormal basis $\{N_1, \ldots, N_{n-d}\}$ of $T_\eta \mathcal{M}^\perp$.
8: Compute $N = \sum_{i=1}^{n-d} \zeta^i N_i$.
9: **for** $j = 1, \ldots, d$ **do**
10:   **for** $k = 1, \ldots, d$ **do**
11:     Compute $\tilde{L}_j^k = \sum_{l=1}^{d} g^{kl} \langle N, t_{jl} \rangle$.
12:   **end for**
13: **end for**
14: Set $L_\zeta = [\tilde{L}_j^k]_{j,k=1,\ldots,d}$.
15: Compute the maximum eigenvalue λ_ζ of L_ζ.

with the second fundamental form. According to the standard definition (Carmo 1992, Proposition 2.3),

$$L_\zeta(X) = -(\nabla_X N)^T, \quad (2)$$

where $X \in T_\eta \mathcal{M}$ and ∇_X denotes the covariant derivative in $\mathbb{R}^n$. Moreover, $(\,\cdot\,)^T$ denotes the projection on the tangent space. It can be shown that the linear operator L_ζ has the following matrix representation: $L_\zeta = [\tilde{L}_j^k]_{j,k=1,\ldots,d}$ where $\tilde{L}_j^k = \sum_{l=1}^{d} g^{kl} \langle N, t_{jl} \rangle$ and $t_{ij} = \frac{\partial^2 s}{\partial \eta_i \partial \eta_j}$, $i, j = 1, \ldots, d$, denote the mixed second order partial derivatives of s with respect to η. We refer to (Kokiopoulou, Kressner, and Frossard) for details. It is important to mention that the operator L_ζ is self-adjoint with respect to the induced metric in the tangent space (Carmo 1992) and therefore its eigenvalues are real. The maximum eigenvalue of L_ζ is usually called the *principal curvature*. Algorithm 1 provides an efficient numerical procedure for computing the principal curvature at a point $s(\eta)$ on the manifold along a certain normal direction ζ.

Application to manifold discretization

We consider a facial image s undergoing rotations θ and isotropic scaling α; that is, the transformation parameter space $\mathcal{H}$ is $(\theta, \alpha) \in [0, 2\pi) \times [0.5, 1.5]$. Our goal is to *construct* a k-NN graph, whose graph nodes correspond to transformed faces $s(\theta, \alpha)$, such that the graph provides a sensible discrete model of the manifold in the sense that graph neighbors represent close-by transformations. The graph construction problem is then equivalent to manifold discretization, since a sample (θ, α) in $\mathcal{H}$ becomes a graph node. When $\mathcal{H}$ is discretized uniformly with $N_\theta = 10$ samples over θ and $N_\alpha = 11$ samples over α, one gets the k-NN graph shown in Fig. 2(a) for $k = 4$, where the Euclidean distances are computed in the ambient space among the $s(\eta)$. Observe the problematic areas where small transformations of the image might not appear to be within nearest neighbors. This is represented by the presence of "distant" edges.

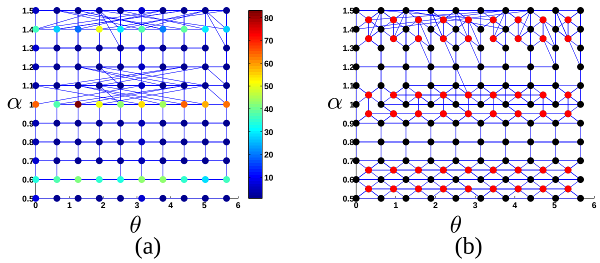

Figure 2: (a) k-NN graph with uniform discretization (color shading depicts the curvature values). (b) k-NN graph with curvature-aware discretization (the red nodes correspond to the newly added nodes).

Also, the absence of "short" edges in the bottom area implies that nearest neighbors are found only along the horizontal (θ) direction. Thus, such a graph obtained from uniform discretization in the parameter space has troubles capturing the manifold geometry. The color shading of the nodes in Fig. 2(a) represents the manifold curvature, computed at each node by sampling 40 random normal directions ζ and reporting the maximum value. We used bilinear interpolation and finite differences for computing the derivatives in steps 3 and 4 of Algorithm 1. One can see that the problematic areas are readily identified by the curvature values. This implies that one may design a curvature-aware manifold discretization approach, where more data samples will be generated in highly curved areas. As an example, we show in Fig. 2(b) the k-NN graph ($k = 4$) obtained when the higher resolution neighbors (red nodes) of the samples with the highest curvature values are included in the original dataset (black nodes). Observe that now the number of "distant" edges has been significantly reduced and the new graph provides a more consistent discrete model of the (continuous) manifold. Of course, the number of nodes has increased and this example may not be directly compared to the first one. The main point is that adding samples in the areas of high curvature can help towards constructing a consistent graph. In our future work, we plan to formalize this intuition and design a solid curvature-aware algorithm for manifold discretization.

References

Baraniuk, R. G., and Wakin, M. B. 2009. Random projections of smooth manifolds. *Foundations of Computational Mathematics* 9(1):51–77.

Carmo, M. P. D. 1992. *Riemannian Geometry*. Birkhäuser.

Kokiopoulou, E.; Kressner, D.; and Frossard, P. 2009. Optimal image alignment with random projections of manifolds: algorithm and geometric analysis. *SAM report 2009-41*.

Kokiopoulou, E.; Pirillos, S.; and Frossard, P. 2008. Graph-based classification for multiple observations of transformed patterns. *IEEE Int. Conf. Pattern Recognition (ICPR)*.

Pozdnoukhov, A., and Bengio, S. 2006. Graph-based transformation manifolds for invariant pattern recognition with kernel methods. *IEEE Int. Conf. on Pattern Recognition (ICPR)*.