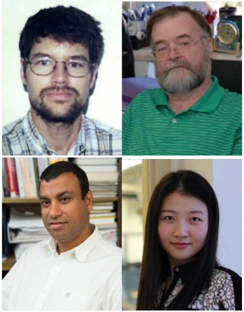
Riemannian Optimization and its Application to Elastic Shape Analysis

Wen Huang

Université catholique de Louvain

Joint work with:

- Pierre-Antoine Absil, Professor of Mathematical Engineering, *Université catholique de Louvain*
- Kyle A. Gallivan, Professor of Mathematics, *Florida State University*
- Anuj Srivastava, Professor of Statistics, *Florida State University*
- Yaqing You, Ph.D candidate in Applied and Computational Mathematics, *Florida State University*

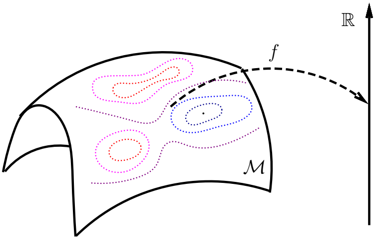


Riemannian Optimization

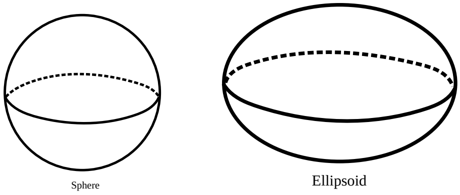
Problem: Given $f(x) : \mathcal{M} \rightarrow \mathbb{R}$, solve

$$\min_{x \in \mathcal{M}} f(x)$$

where $\mathcal{M}$ is a Riemannian manifold.



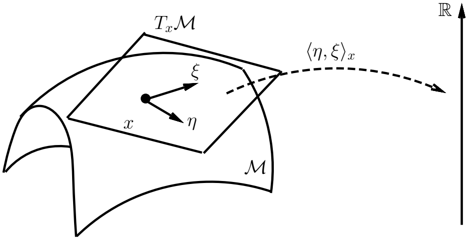
Examples of Manifolds



- Stiefel manifold: $\mathrm{St}(p, n) = \{X \in \mathbb{R}^{n \times p} | X^T X = I_p\}$
- Grassmann manifold: Set of all p-dimensional subspaces of $\mathbb{R}^n$
- Set of fixed rank m-by-n matrices
- And many more

Riemannian Manifolds

Roughly, a Riemannian manifold $\mathcal{M}$ is a smooth set with a smoothly-varying inner product on the tangent spaces.



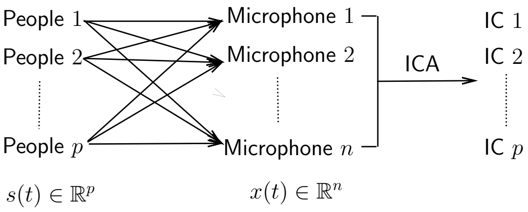
Applications

Four applications are used to demonstrate the importances of the Riemannian optimization:

- Independent component analysis [CS93]
- Matrix completion problem [Van12]
- Geometric mean of symmetric positive definite matrices [ALM04, JVV12, CS15]
- Elastic shape analysis of curves [SKJJ11, HGSA15]

Application: Independent Component Analysis

Cocktail party problem



- Observed signal is $x(t) = As(t)$
- One approach:
  - Assumption: $E\{s(t)s(t+\tau)\}$ is diagonal for all τ
  - $C_\tau(x) := E\{x(t)x(x+\tau)^T\} = AE\{s(t)s(t+\tau)^T\}A^T$

Application: Independent Component Analysis

- Minimize joint diagonalization cost function on the Stiefel manifold [TI06]:

$$f : \mathrm{St}(p, n) \to \mathbb{R} : V \mapsto \sum_{i=1}^{N} \|V^T C_i V - \mathrm{diag}(V^T C_i V)\|_F^2.$$

- $C_1, \dots, C_N$ are covariance matrices and $\mathrm{St}(p, n) = \{X \in \mathbb{R}^{n \times p} | X^T X = I_p\}$.

Application: Matrix Completion Problem

Matrix completion problem

$$\begin{array}{cc} & \begin{matrix}\text{Movie } 1 & \text{Movie } 2 & & & & \text{Movie } n\end{matrix} \\ \begin{matrix}\text{User } 1 \\ \text{User } 2 \\ \\ \text{User } m\end{matrix} & \begin{pmatrix} & 1 & & & 4 & \\ 3 & 5 & & & 4 & \\ & & 5 & 1 & & \\ & 2 & & & 5 & 3 \end{pmatrix} \end{array}$$

Rate matrix M

- The matrix M is sparse
- The goal: complete the matrix M

Application: Matrix Completion Problem

$$
\begin{array}{ccc}
\text{movies} & \text{meta-user} & \text{meta-movie} \\
\begin{pmatrix}
a_{11} & & & a_{14} \\
 & & & a_{24} \\
 & & a_{33} & \\
a_{41} & & & \\
 & a_{52} & a_{53} &
\end{pmatrix}
=
&
\begin{pmatrix}
b_{11} & b_{12} \\
b_{21} & b_{22} \\
b_{31} & b_{32} \\
b_{41} & b_{42} \\
b_{51} & b_{52}
\end{pmatrix}
&
\begin{pmatrix}
c_{11} & c_{12} & c_{13} & c_{14} \\
c_{21} & c_{22} & c_{23} & c_{24}
\end{pmatrix}
\end{array}
$$

- Minimize the cost function

$$f : \mathbb{R}_r^{m \times n} \to \mathbb{R} : X \mapsto f(X) = \|P_\Omega M - P_\Omega X\|_F^2.$$

- $\mathbb{R}_r^{m \times n}$ is the set of m-by-n matrices with rank r. It is known to be a Riemannian manifold.

Application: Geometric Mean of Symmetric Positive Definite (SPD) Matrices

Computing the mean of a population of SPD matrices is important in medical imaging, image processing, radar signal processing, and elasticity. The desired properties are given in the ALM[1] list, some of which are

- if $A_1, \ldots, A_k$ commute, then $G(A_1, \ldots, A_k) = (A_1 \ldots A_k)^{\frac{1}{k}}$;
- $G(A_{\pi(1)}, \ldots, A_{\pi(k)}) = G(A_1, \ldots, A_k)$, with π a permutation of $(1, \ldots, k)$;
- $G(A_1, \ldots, A_k) = G\left(A_1^{-1}, \ldots A_k^{-1}\right)^{-1}$;
- $\det G(A_1, \ldots, A_k) = (\det A_1 \ldots \det A_k)^{\frac{1}{k}}$;

where $A_1, \ldots, A_k$ are SPD matrices, and $G(\cdot, \ldots, \cdot)$ denotes the geometric mean of arguments.

[1] T. Ando, C.-K. Li, and R. Mathias, Geometric means, *Linear Algebra and Its Applications*, 385:305-334, 2004

Application: Geometric Mean of Symmetric Positive Definite Matrices

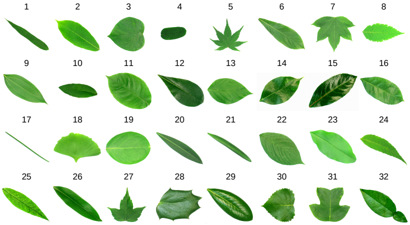
One geometric mean is the Karcher mean of the manifold of SPD matrices with the affine invariant metric, i.e.,

$$G(A_1, \ldots, A_k) = \arg \min_{X \in \mathrm{S}_+^n} \frac{1}{2k} \sum_{i=1}^{k} \mathrm{dist}^2(X, A_i),$$

where $\mathrm{dist}(X, Y) = \| \log(X^{-1/2} Y X^{-1/2}) \|_F$ is the distance under the Riemannian metric

$$g(\eta_X, \xi_X) = \mathrm{trace}(\eta_X X^{-1} \xi_X X^{-1}).$$

Application: Elastic Shape Analysis of Curves



- Classification [LKS+12, HGSA15]
- Face recognition [DBS+13]

Application: Elastic Shape Analysis of Curves

- Elastic shape analysis invariants:
  - Rescaling
  - Translation
  - Rotation
  - Reparametrization
- The shape space is a quotient space

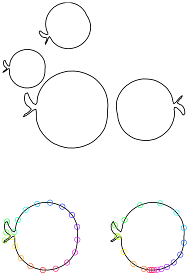
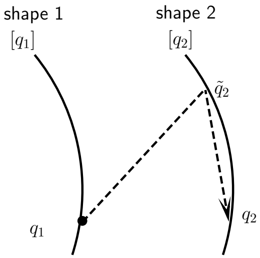


Figure: All are the same shape.

Application: Elastic Shape Analysis of Curves



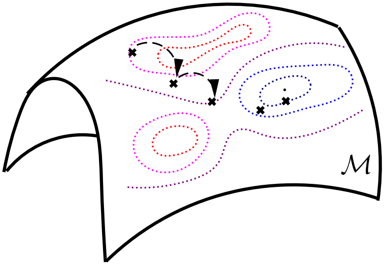
- Optimization problem $\min_{q_2 \in [q_2]} \mathrm{dist}(q_1, q_2)$ is defined on a Riemannian manifold
- Computation of a geodesic between two shapes
- Computation of Karcher mean of a population of shapes

More Applications

- Large-scale Generalized Symmetric Eigenvalue Problem and SVD
- Blind source separation on both Orthogonal group and Oblique manifold
- Low-rank approximate solution symmetric positive definite Lyapanov $AXM + MXA = C$
- Best low-rank approximation to a tensor
- Rotation synchronization
- Graph similarity and community detection
- Low rank approximation to role model problem

Comparison with Constrained Optimization

- All iterates on the manifold
- Convergence properties of unconstrained optimization algorithms
- No need to consider Lagrange multipliers or penalty functions
- Exploit the structure of the constrained set



Iterations on the Manifold

Consider the following generic update for an iterative Euclidean optimization algorithm:

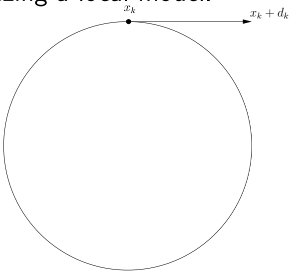
$$x_{k+1} = x_k + \Delta x_k = x_k + \alpha_k s_k \ .$$

This iteration is implemented in numerous ways, e.g.:

- Steepest descent: $x_{k+1} = x_k - \alpha_k \nabla f(x_k)$
- Newton's method: $x_{k+1} = x_k - \left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k)$
- Trust region method: Δx_k is set by optimizing a local model.

Objects

- Direction/movement: $s_k / \Delta x_k$
- Gradient: $\nabla f(x_k)$
- Hessian: $\nabla^2 f(x_k)$
- Addition: $+$



Riemannian gradient and Riemannian Hessian

Definition

The Riemannian gradient of f at x is the unique tangent vector in $T_x M$ satisfying $\forall \eta \in T_x M$, the directional derivative

$$\mathrm{D}\, f(x)[\eta] = \langle \operatorname{grad} f(x), \eta \rangle$$

and $\operatorname{grad} f(x)$ is the direction of steepest ascent.

Definition

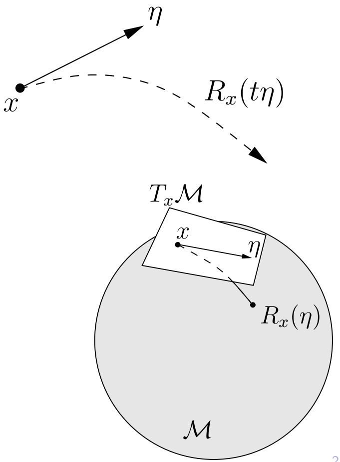
The Riemannian Hessian of f at x is a symmetric linear operator from $T_x M$ to $T_x M$ defined as

$$\operatorname{Hess} f(x) : T_x M \to T_x M : \eta \to \nabla_\eta \operatorname{grad} f,$$

where ∇ is the affine connection.

Retractions

Euclidean	Riemannian
$x_{k+1} = x_k + \alpha_k d_k$	$x_{k+1} = R_{x_k}(\alpha_k \eta_k)$



Definition

A retraction is a mapping R from TM to M satisfying the following:

- R is continuously differentiable
- $R_x(0) = x$
- $\mathrm{D}\, R_x(0)[\eta] = \eta$

- maps tangent vectors back to the manifold
- defines curves in a direction

Categories of Riemannian optimization methods

Retraction-based: local information only

Line search-based: use local tangent vector and $R_x(t\eta)$ to define line

- Steepest decent
- Newton

Local model-based: series of flat space problems

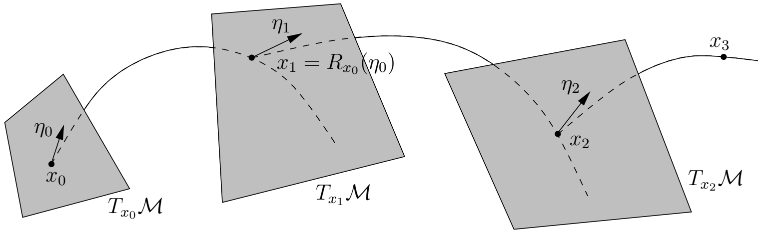
- Riemannian trust region Newton (RTR)
- Riemannian adaptive cubic overestimation (RACO)

Generic Riemannian Optimization Algorithm

1. At iterate $x \in M$
2. Find $\eta \in T_x M$ which satisfies certain condition.
3. Choose new iterate $x_+ = R_x(\eta)$.
4. Goto step 1.

A suitable setting

This paradigm is sufficient for describing many optimization methods.



Categories of Riemannian optimization methods

Elements required for optimizing a cost function (M, g):

- an representation for points x on M, for tangent spaces $T_x M$, and for the inner products $g_x(\cdot, \cdot)$ on $T_x M$;
- choice of a retraction $R_x : T_x M \to M$;
- formulas for $f(x), \mathrm{grad}\ f(x)$ and $\mathrm{Hess}\ f(x)$ (or its action);
- Computational and storage efficiency;

Categories of Riemannian optimization methods

Retraction and transport-based: information from multiple tangent spaces

- Conjugate gradient: multiple tangent vectors
- Quasi-Newton e.g. Riemannian BFGS: transport operators between tangent spaces

Additional element required for optimizing a cost function (M, g):

- formulas for combining information from multiple tangent spaces.

Vector Transports

Vector Transport

- Vector transport: Transport a tangent vector from one tangent space to another
- $\mathcal{T}_{\eta_x}\xi_x$, denotes transport of ξ_x to tangent space of $R_x(\eta_x)$. R is a retraction associated with $\mathcal{T}$
- Isometric vector transport $\mathcal{T}_{\mathrm{S}}$ preserve the length of tangent vector

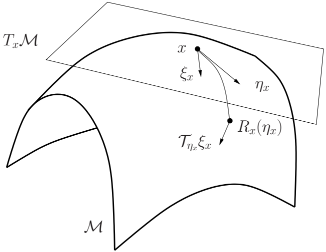


Figure: Vector transport.

Retraction/Transport-based Riemannian Optimization

Benefits

- Increased generality does not compromise the important theory
- Less expensive than or similar to previous approaches
- May provide theory to explain behavior of algorithms specifically developed for a particular application – or closely related ones

Possible Problems

- May be inefficient compared to algorithms that exploit application details

Some History of Optimization On Manifolds (I)

Luenberger (1973), *Introduction to linear and nonlinear programming*. Luenberger mentions the idea of performing line search along geodesics, "which we would use if it were computationally feasible (which it definitely is not)". Rosen (1961) essentially anticipated this but was not explicit in his Gradient Projection Algorithm.

Gabay (1982), *Minimizing a differentiable function over a differential manifold*. Steepest descent along geodesics; Newton's method along geodesics; Quasi-Newton methods along geodesics. On Riemannian submanifolds of $\mathbb{R}^n$.

Smith (1993-94), *Optimization techniques on Riemannian manifolds*. Levi-Civita connection ∇; Riemannian exponential mapping; parallel translation.

Some History of Optimization On Manifolds (II)

The "pragmatic era" begins:

Manton (2002), *Optimization algorithms exploiting unitary constraints* "The present paper breaks with tradition by not moving along geodesics". The geodesic update $\mathrm{Exp}_x \eta$ is replaced by a projective update $\pi(x + \eta)$, the *projection* of the point $x + \eta$ onto the manifold.

Adler, Dedieu, Shub, et al. (2002), *Newton's method on Riemannian manifolds and a geometric model for the human spine*. The exponential update is relaxed to the general notion of *retraction*. The geodesic can be replaced by any (smoothly prescribed) curve tangent to the search direction.

Absil, Mahony, Sepulchre (2007) Nonlinear conjugate gradient using retractions.

Some History of Optimization On Manifolds (III)

Theory, efficiency, and library design improve dramatically:

Absil, Baker, Gallivan (2004-07), Theory and implementations of Riemannian Trust Region method. Retraction-based approach. Matrix manifold problems, software repository

http://www.math.fsu.edu/~cbaker/GenRTR

Anasazi Eigenproblem package in Trilinos Library at Sandia National Laboratory

Absil, Gallivan, Qi (2007-10), Basic theory and implementations of Riemannian BFGS and Riemannian Adaptive Cubic Overestimation. Parallel translation and Exponential map theory, Retraction and vector transport empirical evidence.

Some History of Optimization On Manifolds (IV)

Ring and Wirth (2012), combination of differentiated retraction and isometric vector transport for convergence analysis of RBFGS

Absil, Gallivan, Huang (2009-2015), Complete theory of Riemannian Quasi-Newton and related transport/retraction conditions, Riemannian SR1 with trust-region, RBFGS on partly smooth problems, A C++ library: `http://www.math.fsu.edu/~whuang2/ROPTLIB`

Sato, Iwai (2013-2015), Global convergence analysis using the differentiated retraction for Riemannian conjugate gradient methods

Many people Application interests start to increase noticeably

Current UCL/FSU Methods

- Riemannian Steepest Descent
- Riemannian Trust Region Newton: global, quadratic convergence
- Riemannian Broyden Family : global (convex), superlinear convergence
- Riemannian Trust Region SR1: global, $(d+1)-$superlinear convergence
- For large problems
  - Limited memory RTRSR1
  - Limited memory RBFGS
- Riemannian conjugate gradient (much more work to do on local analysis)
- A library is available at `www.math.fsu.edu/~whuang2/ROPTLIB`

Current/Future Work on Riemannian methods

- Manifold and inequality constraints
- Discretization of infinite dimensional manifolds and the convergence/accuracy of the approximate minimizers – specific to a problem and extracting general conclusions
- Partly smooth cost functions on Riemannian manifold

Elastic Shape Analysis of Curves

- Elastic shape analysis invariants
  - Rescaling
  - Translation
  - Rotation
  - Reparameterization (difficult)

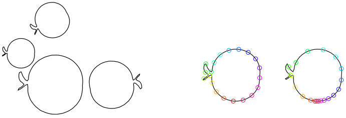
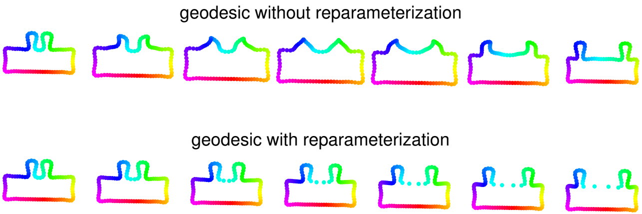


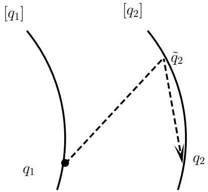
Figure: All are the same shape.



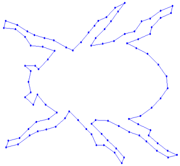
Best Rotation and Reparameterization

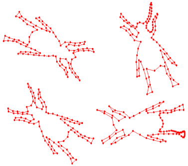
$$(O_*, \gamma_*) = \operatorname*{argmin}_{(O,\gamma)\in \mathrm{SO}(n)\times\Gamma} \mathrm{dist}_{l_n}(q_1, O(q_2, \gamma)),$$

where $\mathrm{SO}(n)$ is the orthogonal group and Γ is the set of absolutely continuous bijection from $\mathbb{S}^1$ to $\mathbb{S}^1$.



Updating O and γ





Optimization Algorithms

- Coordinate Descent Method: Optimize rotation and reparameterization alternately.
  - Rotation: Procrustes problem solved using SVD
  - Reparameterization: $O(N)$ runs of Dynamic programming (DP) with slope constraints, where N is the number of points in the curves
  - Complexity is $O(N^3)$ per iteration.
- Riemannian Method
  - Domain: $\mathrm{SO}(n) \times \mathbb{R} \times \mathbb{S}^{\mathbb{L}^2}$, where $\mathbb{S}^{\mathbb{L}^2}$ is the unit sphere in $\mathbb{L}^2$.
  - Complexity is $O(N)$ per iteration.
- A global minimizer is desired

Examples (by Riemannian methods)

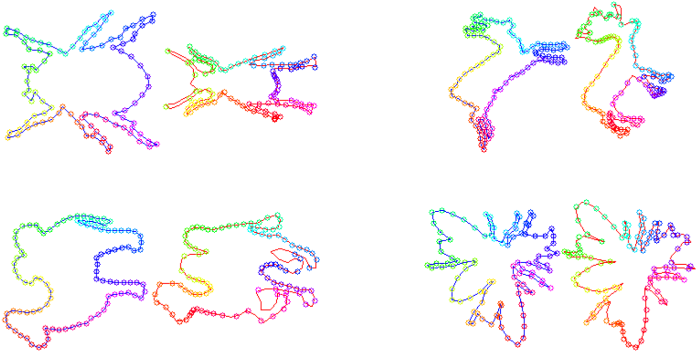
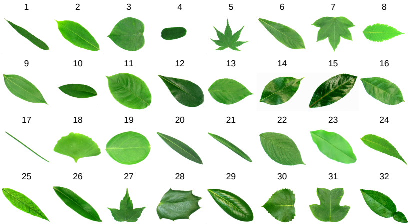


Figure: Applying best rotation and reparameterization to one of the curves. The colors indicate corresponding points on the two curves.

Data Sets

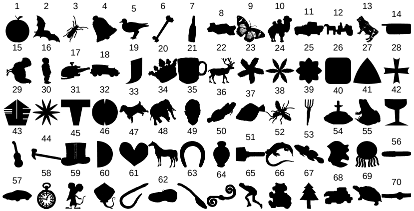
Flavia leaf dataset [WBX+07]

- 1907 images of leaves
- 32 species



MPEG-7 dataset [Uni]

- 1400 binary images
- 70 clusters



- Boundary curves: BWBOUNDARIES function in Matlab
- 100 points in $\mathbb{R}^2$ used for each boundary

Known $\gamma_T^{-1}(t) = (t + \sin(2\pi t))/(4\pi)$

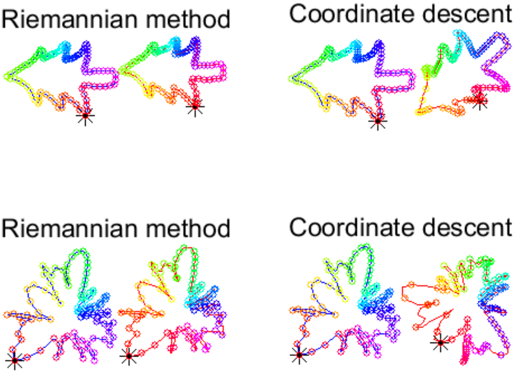


Figure: Apply random rotation and given γ_T^{-1} to a given shape to obtain the second shape. For the tested 1020 shapes, coordinate descent method may not find a global minimizer.

One Nearest Neighbor Results

- The 1NN metric, μ, computes the percentage of points whose nearest neighbor are in the same cluster, i.e.,

$$\mu = \frac{1}{n}\sum_{i=1}^{n} C(i), \quad C(i) = \begin{cases} 1 & \text{if point } i \text{ and its nearest neighbor} \\ & \text{are in the same cluster;} \\ 0 & \text{otherwise.} \end{cases}$$

	t_{ave}(F)	1NN(F)	t_{ave}(M)	1NN(M)
Riemannian methods	0.088	89.51%	0.181	97.79%
Coordinate descent	0.897	87.52%	0.908	96.79%

Geodesic

The energy function is

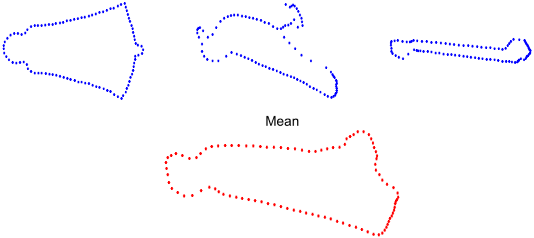
$$E : \mathcal{P}_{q_1,[q_2]} \to \mathbb{R} : \alpha \mapsto \frac{1}{2} \int_0^1 \langle \dot{\alpha}(\tau), \dot{\alpha}(\tau) \rangle \, d\tau,$$

where $\mathcal{P}_{q_1,[q_2]}$ denotes the set of paths connecting q_1 and $q \in [q_2]$.

Karcher Mean

The Karcher mean of shapes $[q_i], i = 1, 2, \ldots, N$ is defined to be the minimizer of the cost function

$$[q_*] = \arg\min_{[q]} \frac{1}{2N} \sum_{i=1}^{N} \mathrm{dist}^2([q], [q_i]).$$



Summary

- Introduced the framework of Riemannian optimization and the state-of-the-art Riemannian algorithms
- Used applications to show the importance of Riemannian optimization
- Introduced the framework of elastic shape analysis of curves and showed the performance of Riemannian optimization in this application

Thanks!

References I

T. Ando, C. K. Li, and R. Mathias.
Geometric means.
Linear Algebra and Its Applications, 385:305–334, 2004.

J. F. Cardoso and A. Souloumiac.
Blind beamforming for non-gaussian signals.
IEE Proceedings F Radar and Signal Processing, 140(6):362, 1993.

A. Cherian and S. Sra.
Riemannian dictionary learning and sparse coding for positive definite matrices.
CoRR, abs/1507.02772, 2015.

H. Drira, B. Ben Amor, A. Srivastava, M. Daoudi, and R. Slama.
3D face recognition under expressions, occlusions, and pose variations.
Pattern Analysis and Machine Intelligence, IEEE Transactions on, 35(9):2270–2283, 2013.

Wen Huang, K. A. Gallivan, Anuj Srivastava, and P.-A. Absil.
Riemannian optimization for registration of curves in elastic shape analysis.
Journal of Mathematical Imaging and Vision, 2015.
DOI:10.1007/s10851-015-0606-8.

B. Jeuris, R. Vandebril, and B. Vandereycken.
A survey and comparison of contemporary algorithms for computing the matrix geometric mean.
Electronic Transactions on Numerical Analysis, 39:379–402, 2012.

References II

H. Laga, S. Kurtek, A. Srivastava, M. Golzarian, and S. J. Miklavcic.
A Riemannian elastic metric for shape-based plant leaf classification.
2012 International Conference on Digital Image Computing Techniques and Applications (DICTA), pages 1–7, December 2012.
doi:10.1109/DICTA.2012.6411702.

A. Srivastava, E. Klassen, S. H. Joshi, and I. H. Jermyn.
Shape analysis of elastic curves in Euclidean spaces.
IEEE Transactions on Pattern Analysis and Machine Intelligence, 33(7):1415–1428, September 2011.
doi:10.1109/TPAMI.2010.184.

F. J. Theis and Y. Inouye.
On the use of joint diagonalization in blind signal processing.
2006 IEEE International Symposium on Circuits and Systems, (2):7–10, 2006.

Temple University.
Shape similarity research project.
www.dabi.temple.edu/ shape/MPEG7/dataset.html.

B. Vandereycken.
Low-rank matrix completion by Riemannian optimization—extended version.
SIAM Journal on Optimization, 23(2):1214–1236, 2012.

S. G. Wu, F. S. Bao, E. Y. Xu, Y.-X. Wang, Y.-F. Chang, and Q.-L. Xiang.
A leaf recognition algorithm for plant classification using probabilistic neural network.
2007 IEEE International Symposium on Signal Processing and Information Technology, pages 11–16, 2007.
arXiv:0707.4289v1.