

Optimization On Manifolds

Pierre-Antoine Absil

Based on P.-A. Absil, R. Mahony, R. Sepulchre, "Optimization Algorithms on Matrix Manifolds", Princeton University Press, January 2008

Compiled on June 29, 2019

Outline

First glimpse

Intro

Overview of application to eigenvalue problem

Manifolds, submanifolds, quotient manifolds

Steepest descent

Newton

Rayleigh on Grassmann

Trust-Region Methods

Vector Transport

BFGS on manifolds

News flashes

Data fitting on manifolds

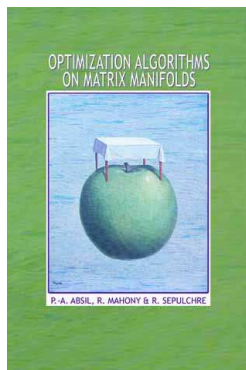
Manifolds in power system management

Conclusion

Collaborations

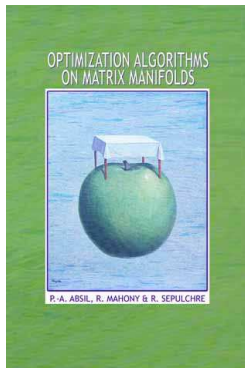
- ▶ Chris Baker, Wen Huang, Kyle Gallivan (Florida State University)
- ▶ Nicolas Boumal, Paul Van Dooren (Université catholique de Louvain)
- ▶ Several other colleagues mentioned later on

Reference



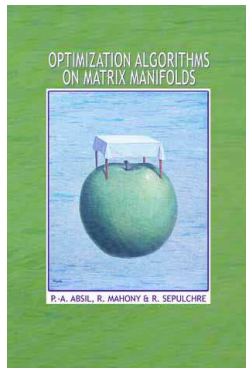
Optimization Algorithms on Matrix Manifolds
P.-A. Absil, R. Mahony, R. Sepulchre
Princeton University Press, January 2008

About the reference



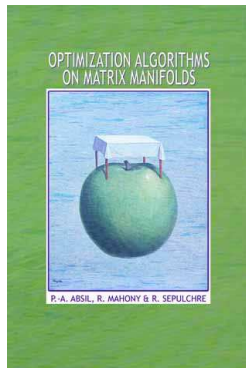
- ▶ The publisher, Princeton University Press, has been a non-profit company since 1910.
- ▶ PDF version of book chapters available on the publisher's web site.

Reference: contents



1. Introduction
2. Motivation and applications
3. Matrix manifolds: first-order geometry
4. Line-search algorithms
5. Matrix manifolds: second-order geometry
6. Newton's method
7. Trust-region methods
8. A constellation of superlinear algorithms

Matrix Manifolds: first-order geometry



Chap 3: Matrix Manifolds: first-order geometry

1. Charts, atlases, manifolds
2. Differentiable functions
3. Embedded submanifolds
4. Quotient manifolds
5. Tangent vectors and differential maps
6. Riemannian metric, distance, gradient

A toolbox

<http://www.manopt.org/>

Manopt

Home

Tutorial

Forum

About

Contact

Welcome to Manopt!

A Matlab toolbox for optimization on manifolds

Optimization on manifolds is a powerful paradigm to address nonlinear optimization problems. With Manopt, it is easy to deal with various types of constraints which arise naturally in applications, such as orthonormality or low rank.

Download

Get started

Ref: Nicolas Boumal et al, *Manopt, a Matlab toolbox for optimization on manifolds*, JMLR 15(Apr) 1455-1459, 2014.

Other generic software for optimization on manifolds

- ▶ ROPTLIB (C++)
<http://www.math.fsu.edu/~whuang2/ROPTLIB.htm>
- ▶ Pymanopt (Python)
<https://pymanopt.github.io/>
- ▶ See also Geomstats (Python package for Riemannian geometry in machine learning)
<https://github.com/geomstats/geomstats>

Manifolds everywhere

Blind source separation – Input



Manifolds everywhere

Blind source separation – Output



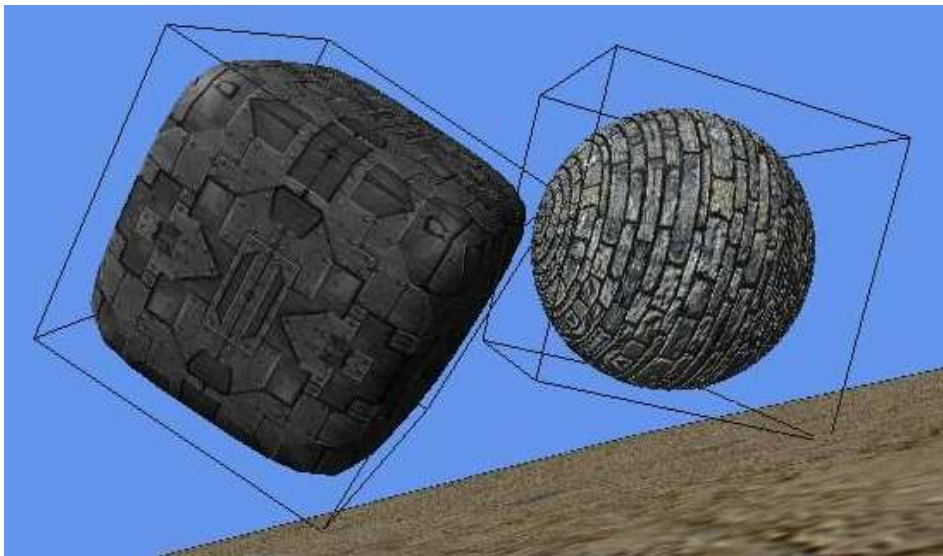
Manifolds everywhere

Vibration analysis

VIDEO: mode1.avi

Manifolds everywhere

Minimum bounding box



Manifolds everywhere

Morphing

VIDEO

Manifolds everywhere

Recommender system

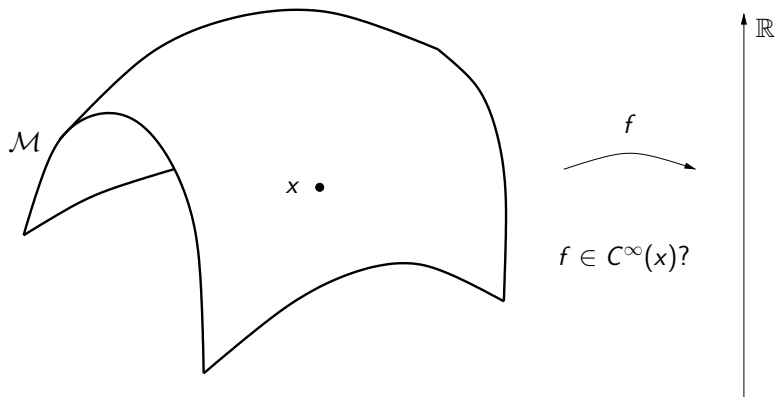
user u

$$X = \begin{pmatrix} 5 & ? & 2 & ? & ? & 5 & ? & ? & 3 & ? & 5 & ? & ? & ? & 2 & ? \\ 3 & ? & 2 & ? & ? & 2 & ? & ? & ? & 3 & 2 & ? & 5 & ? & ? & ? \\ 1 & ? & 5 & 2 & 3 & 4 & ? & 4 & ? & ? & ? & 2 & ? & ? & ? & ? \\ ? & 1 & ? & 3 & ? & ? & ? & 3 & ? & ? & ? & ? & 2 & 1 & 5 & 5 \\ ? & 4 & ? & ? & ? & ? & 5 & ? & ? & ? & 1 & ? & ? & 1 & ? & 4 \end{pmatrix} \text{ movie } i$$



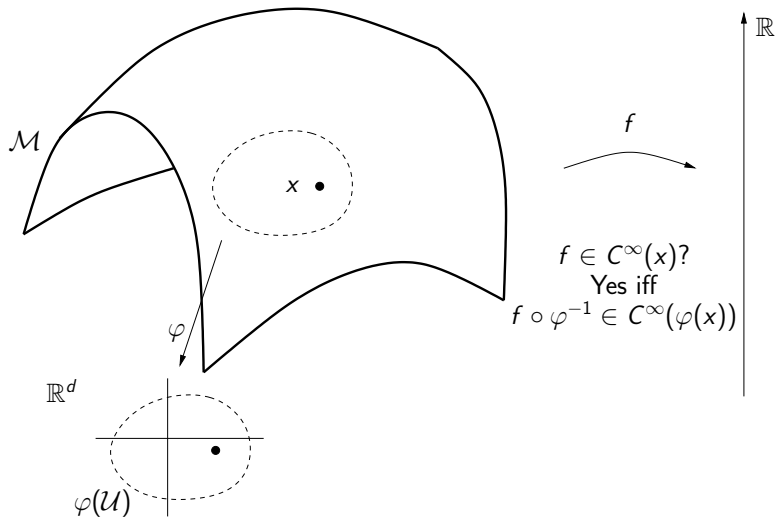
Manifolds

Generalizing \mathbb{R}^n without losing smoothness



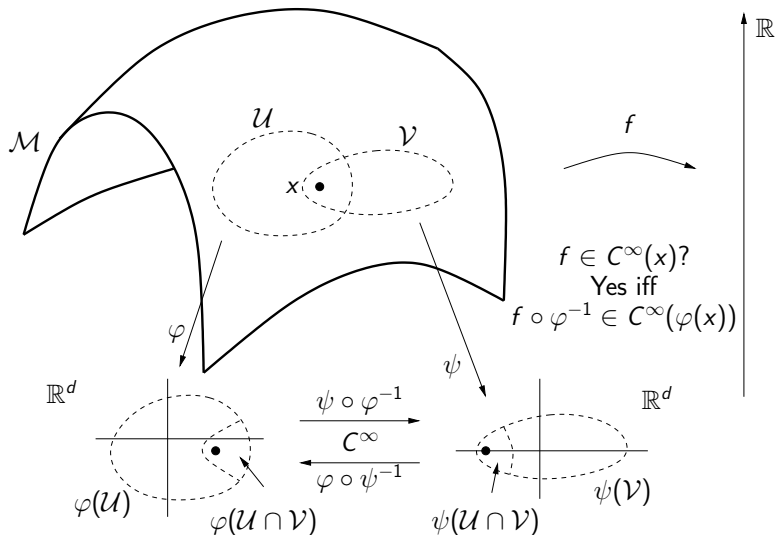
Manifolds

Generalizing \mathbb{R}^n without losing smoothness

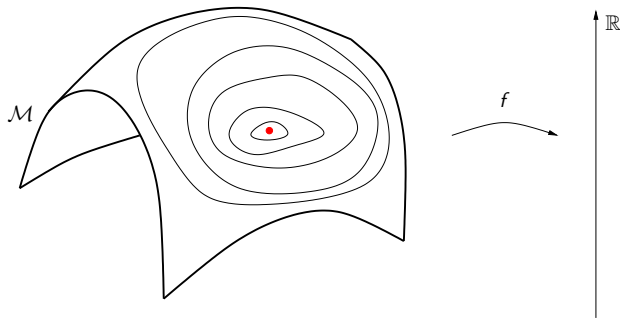


Manifolds

Generalizing \mathbb{R}^n without losing smoothness



Optimization on manifolds: illustration



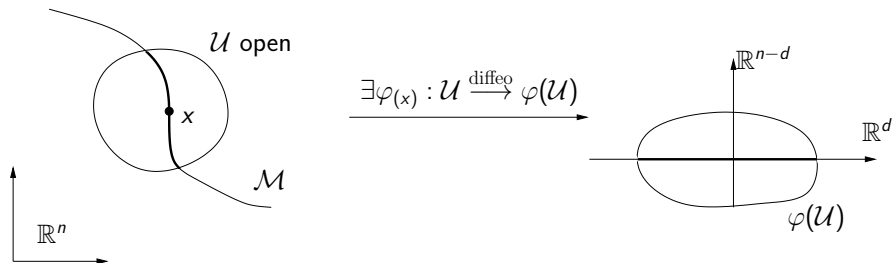
Given:

- ▶ A set \mathcal{M} endowed (explicitly or implicitly) with a manifold structure (i.e., a collection of compatible charts).
- ▶ A function $f : \mathcal{M} \rightarrow \mathbb{R}$, smooth in the sense of the manifold structure.

Task: Compute a minimizer of f .

Manifolds appear in two ways

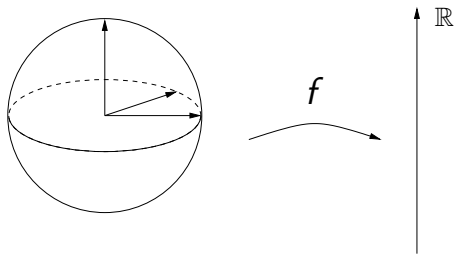
1st way: Submanifolds of \mathbb{R}^n



The set $\mathcal{M} \subset \mathbb{R}^n$ is termed a *submanifold* of \mathbb{R}^n if the situation described above holds for all $x \in \mathcal{M}$.

Manifolds appear in two ways

1st way: Submanifolds of \mathbb{R}^n — for example, the sphere



Manifolds appear in two ways

1st way: Submanifolds of \mathbb{R}^n — for example, the Stiefel manifold

The Stiefel manifold of orthonormal p -frames in \mathbb{R}^n is

$$\text{St}(p, n) = \{X \in \mathbb{R}^{n \times p} : X^T X = I\}.$$

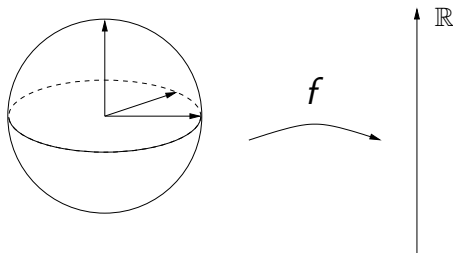
Manifolds appear in two ways

1st way: Submanifolds of \mathbb{R}^n — for example, the Stiefel manifold

The Stiefel manifold of orthonormal p -frames in \mathbb{R}^n is

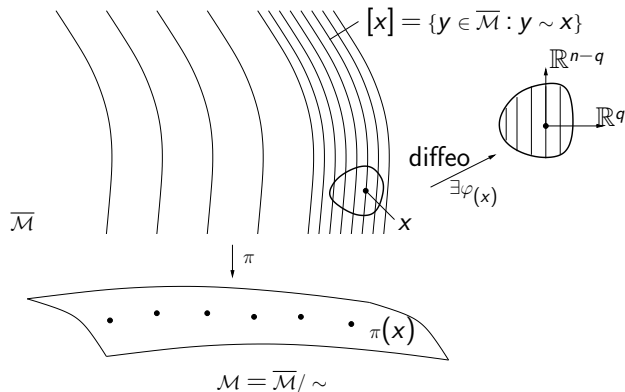
$$\text{St}(p, n) = \{X \in \mathbb{R}^{n \times p} : X^T X = I\}.$$

When $p = 1$, we recover the sphere:



Manifolds appear in two ways

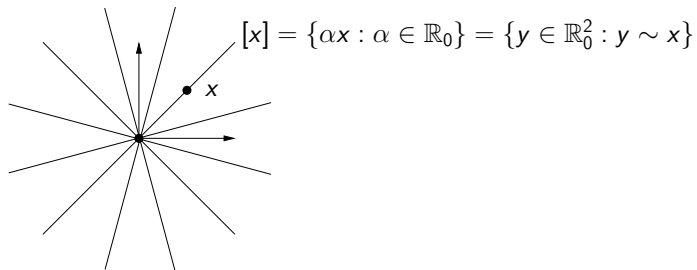
2nd way: Quotient manifold $\overline{\mathcal{M}}/\sim$



The set $\overline{\mathcal{M}}/\sim$ is termed a *quotient manifold* if the situation described above holds for all $x \in \overline{\mathcal{M}}$.

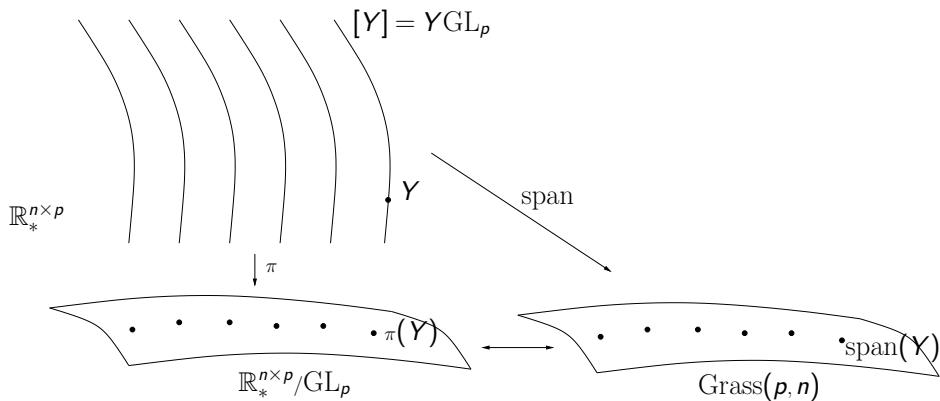
Manifolds appear in two ways

2nd way: Quotient manifold $\overline{\mathcal{M}}/\sim$ — for example, the projective space



Manifolds appear in two ways

2nd way: Quotient manifold $\overline{\mathcal{M}}/\sim$ — for example, the Grassmann manifold



Smooth unconstrained optimization in \mathbb{R}^n

Given: a smooth function

$$f : \mathbb{R}^n \rightarrow \mathbb{R},$$

termed the *cost function* or *objective function*.

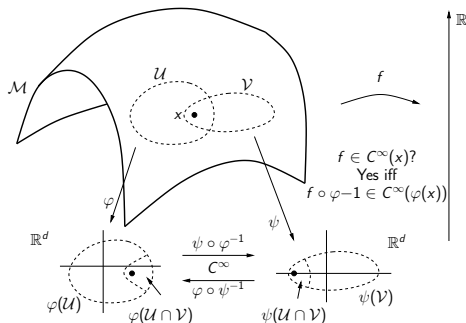
Task: find $x_* \in \mathbb{R}^n$ such that there exists $\epsilon > 0$ for which

$$f(x) \geq f(x_*) \text{ whenever } \|x - x_*\| < \epsilon.$$

Such a point x_* is called a *local minimizer* of f .

Smooth optimization on manifolds

in its most abstract formulation

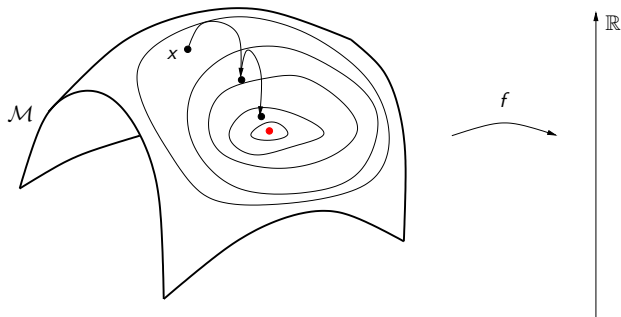


Given:

- ▶ A set \mathcal{M} endowed (explicitly or implicitly) with a manifold structure (i.e., a collection of compatible charts).
- ▶ A function $f : \mathcal{M} \rightarrow \mathbb{R}$, smooth in the sense of the manifold structure.

Task: Compute a local minimizer of f .

Optimization on manifolds: algorithms

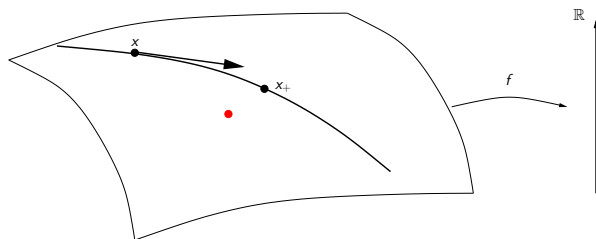


Given:

- ▶ A set \mathcal{M} endowed (explicitly or implicitly) with a manifold structure (i.e., a collection of compatible charts).
- ▶ A function $f : \mathcal{M} \rightarrow \mathbb{R}$, smooth in the sense of the manifold structure.

Task: Compute a local minimizer of f .

Previous work on Optimization On Manifolds



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)”.

The purely Riemannian era

Gabay (1982), *Minimizing a differentiable function over a differential manifold*. Stepest descent along geodesics; Newton's method along geodesics; Quasi-Newton methods along geodesics.

Smith (1994), *Optimization techniques on Riemannian manifolds*.

Levi-Civita connection ∇ ; Riemannian exponential; parallel translation.
But Remark 4.9: If Algorithm 4.7 (Newton's iteration on the sphere for the Rayleigh quotient) is simplified by replacing the exponential update with the update

$$x_{k+1} = \frac{x_k + \eta_k}{\|x_k + \eta_k\|}$$

then we obtain the Rayleigh quotient iteration.

The pragmatic era

Manton (2002), *Optimization algorithms exploiting unitary constraints*
“The present paper breaks with tradition by not moving along geodesics”. The geodesic update $\text{Exp}_x \eta$ 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.

Looking ahead: Newton on abstract manifolds

Required: Riemannian manifold \mathcal{M} ; retraction R on \mathcal{M} ; affine connection ∇ on \mathcal{M} ; real-valued function f on \mathcal{M} .

Iteration $x_k \in \mathcal{M} \mapsto x_{k+1} \in \mathcal{M}$ defined by

1. Solve the Newton equation

$$\text{Hess } f(x_k)\eta_k = -\text{grad } f(x_k)$$

for the unknown $\eta_k \in T_{x_k}\mathcal{M}$, where

$$\text{Hess } f(x_k)\eta_k := \nabla_{\eta_k} \text{grad } f.$$

2. Set

$$x_{k+1} := R_{x_k}(\eta_k).$$

Looking ahead: Newton on submanifolds of \mathbb{R}^n

Required: Riemannian submanifold \mathcal{M} of \mathbb{R}^n ; retraction R on \mathcal{M} ; real-valued function f on \mathcal{M} .

Iteration $x_k \in \mathcal{M} \mapsto x_{k+1} \in \mathcal{M}$ defined by

1. Solve the Newton equation

$$\text{Hess } f(x_k)\eta_k = -\text{grad } f(x_k)$$

for the unknown $\eta_k \in T_{x_k}\mathcal{M}$, where

$$\text{Hess } f(x_k)\eta_k := P_{T_{x_k}\mathcal{M}}D(\text{grad } f)(x_k)[\eta_k].$$

2. Set

$$x_{k+1} := R_{x_k}(\eta_k).$$

Looking ahead: Newton on the unit sphere S^{n-1}

Required: real-valued function f on S^{n-1} .

Iteration $x_k \in S^{n-1} \mapsto x_{k+1} \in S^{n-1}$ defined by

1. Solve the Newton equation

$$\begin{cases} P_{x_k} D(\text{grad } f)(x_k)[\eta_k] = -\text{grad } f(x_k) \\ x^T \eta_k = 0, \end{cases}$$

for the unknown $\eta_k \in \mathbb{R}^n$, where

$$P_{x_k} = (I - x_k x_k^T).$$

2. Set

$$x_{k+1} := \frac{x_k + \eta_k}{\|x_k + \eta_k\|}.$$

Looking ahead: Newton for Rayleigh quotient optimization on unit sphere

Iteration $x_k \in S^{n-1} \mapsto x_{k+1} \in S^{n-1}$ defined by

1. Solve the Newton equation

$$\begin{cases} P_{x_k} A P_{x_k} \eta_k - \eta_k x_k^T A x_k = -P_{x_k} A x_k, \\ x_k^T \eta_k = 0, \end{cases}$$

for the unknown $\eta_k \in \mathbb{R}^n$, where

$$P_{x_k} = (I - x_k x_k^T).$$

2. Set

$$x_{k+1} := \frac{x_k + \eta_k}{\|x_k + \eta_k\|}.$$

Programme

- ▶ Provide background in differential geometry instrumental for algorithmic development
- ▶ Present manifold versions of some classical optimization algorithms: steepest-descent, Newton, conjugate gradients, trust-region methods
- ▶ Show how to turn these abstract geometric algorithms into practical implementations
- ▶ Illustrate several problems that can be rephrased as optimization problems on manifolds.

Some important manifolds

- ▶ Stiefel manifold $St(p, n)$: set of all orthonormal $n \times p$ matrices.
- ▶ Grassmann manifold $Grass(p, n)$: set of all p -dimensional subspaces of \mathbb{R}^n
- ▶ Euclidean group $SE(3)$: set of all rotations-translations
- ▶ Flag manifold, shape manifold, oblique manifold...
- ▶ Several unnamed manifolds

Motivations for the manifold approach to optimization

- ▶ Exploit the submanifold structure of the feasible set.
 - ▶ Feasible iterates.
 - ▶ Sometimes cheaper methods.

Motivations for the manifold approach to optimization

- ▶ Exploit the submanifold structure of the feasible set.
 - ▶ Feasible iterates.
 - ▶ Sometimes cheaper methods.
- ▶ Exploit the quotient manifold structure of the feasible set.
 - ▶ Stronger theoretical convergence results.
 - ▶ Sometimes (much) stronger practical convergence results.

Motivations for the manifold approach to optimization

- ▶ Exploit the submanifold structure of the feasible set.
 - ▶ Feasible iterates.
 - ▶ Sometimes cheaper methods.
- ▶ Exploit the quotient manifold structure of the feasible set.
 - ▶ Stronger theoretical convergence results.
 - ▶ Sometimes (much) stronger practical convergence results.
- ▶ The objective function involves concepts of differential geometry.
 - ▶ For example, in the Riemannian barycenter problem, the objective function uses the Riemannian distance.

Motivations for the manifold approach to optimization

- ▶ Exploit the submanifold structure of the feasible set.
 - ▶ Feasible iterates.
 - ▶ Sometimes cheaper methods.
- ▶ Exploit the quotient manifold structure of the feasible set.
 - ▶ Stronger theoretical convergence results.
 - ▶ Sometimes (much) stronger practical convergence results.
- ▶ The objective function involves concepts of differential geometry.
 - ▶ For example, in the Riemannian barycenter problem, the objective function uses the Riemannian distance.
- ▶ The adequate choice of differential-geometric tools (e.g., Riemannian metric) can speed up algorithms.
 - ▶ The Riemannian metric affects the steepest-descent direction.

A manifold-based approach to the symmetric eigenvalue problem

OPT



EVP

OPT

Opt algorithms

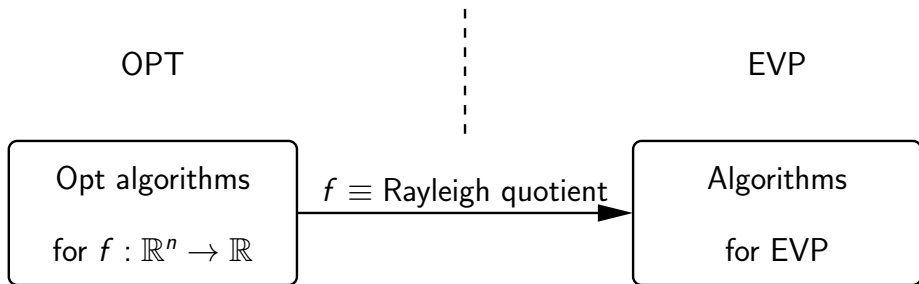
for $f : \mathbb{R}^n \rightarrow \mathbb{R}$

⋮

EVP

Algorithms

for EVP



Rayleigh quotient

Rayleigh quotient of (A, B) :

$$f : \mathbb{R}_*^n \rightarrow \mathbb{R} : f(y) = \frac{y^T A y}{y^T B y}$$

Let A, B in $\mathbb{R}^{n \times n}$, $A = A^T$, $B = B^T \succ 0$,

$$A v_i = \lambda_i B v_i$$

with $\lambda_1 < \lambda_2 \leq \dots \leq \lambda_n$.

Stationary points of f : αv_i , for all $\alpha \neq 0$.

Local (and global) minimizers of f : αv_1 , for all $\alpha \neq 0$.

OPT

Opt algorithms
for $f : \mathbb{R}^n \rightarrow \mathbb{R}$

$f \equiv$ Rayleigh quotient

EVP

Algorithms
for EVP

“Block” Rayleigh quotient

Let $\mathbb{R}_*^{n \times p}$ denote the set of all full-column-rank $n \times p$ matrices.
Generalized (“block”) Rayleigh quotient:

$$f : \mathbb{R}_*^{n \times p} \rightarrow \mathbb{R} : f(Y) = \text{trace} \left((Y^T B Y)^{-1} Y^T A Y \right)$$

Stationary points of f :

$$\begin{bmatrix} v_{i_1} & \dots & v_{i_p} \end{bmatrix} M, \quad \text{for all } M \in \mathbb{R}_*^{p \times p}.$$

Minimizers of f :

$$\begin{bmatrix} v_1 & \dots & v_p \end{bmatrix} M, \quad \text{for all } M \in \mathbb{R}_*^{p \times p}.$$

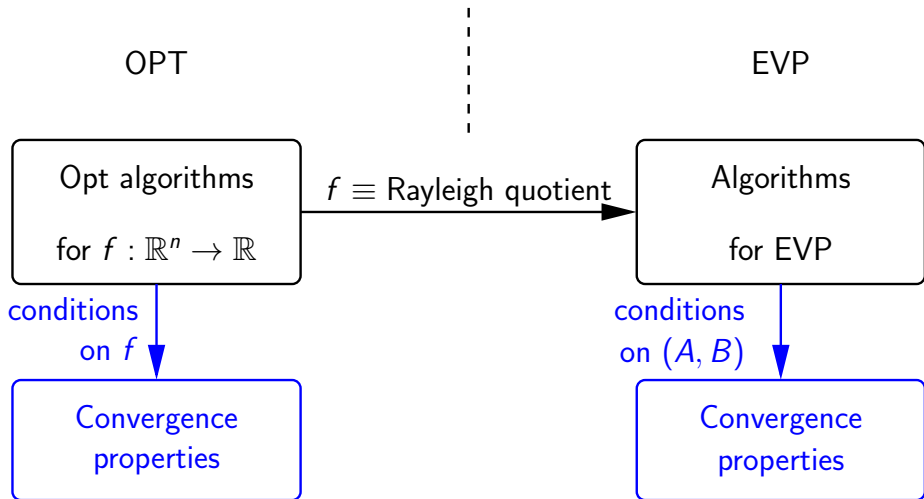
OPT

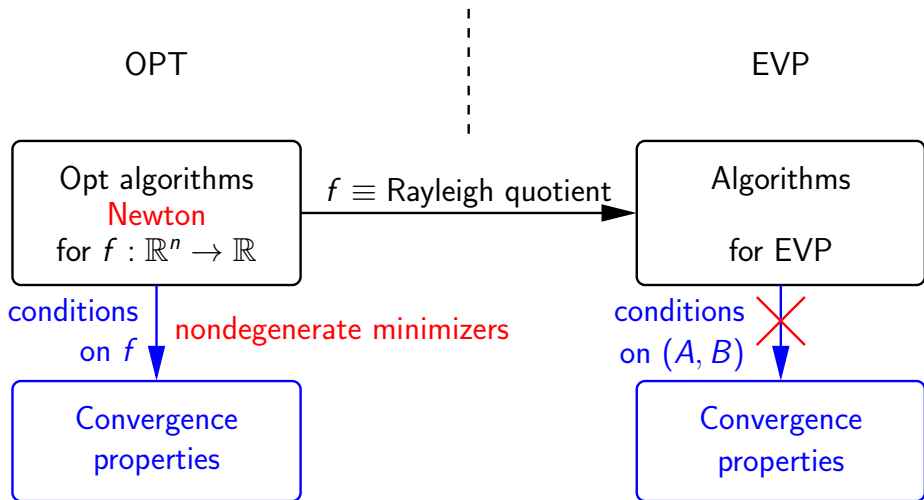
Opt algorithms
for $f : \mathbb{R}^n \rightarrow \mathbb{R}$

$f \equiv$ Rayleigh quotient

EVP

Algorithms
for EVP





Newton for Rayleigh quotient in \mathbb{R}_0^n

Let f denote the Rayleigh quotient of (A, B) .

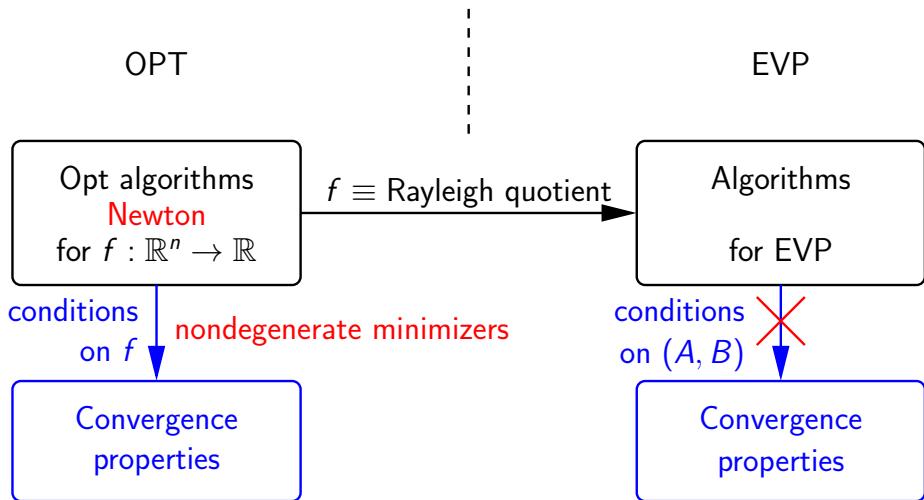
Let $x \in \mathbb{R}_0^n$ be any point such that $f(x) \notin \text{spec}(B^{-1}A)$.

Then the Newton iteration

$$x \mapsto x - (D^2f(x))^{-1} \cdot \text{grad } f(x)$$

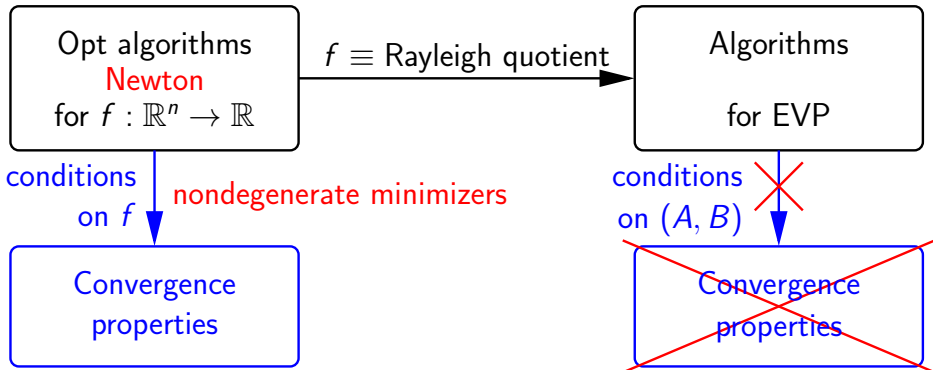
reduces to the iteration

$$x \mapsto 2x.$$



OPT

EVP



Invariance properties of the Rayleigh quotient

Rayleigh quotient of (A, B) :

$$f : \mathbb{R}_*^n \rightarrow \mathbb{R} : f(y) = \frac{y^T A y}{y^T B y}$$

Invariance: $f(\alpha y) = f(y)$ for all $\alpha \in \mathbb{R}_0$.

Invariance properties of the Rayleigh quotient

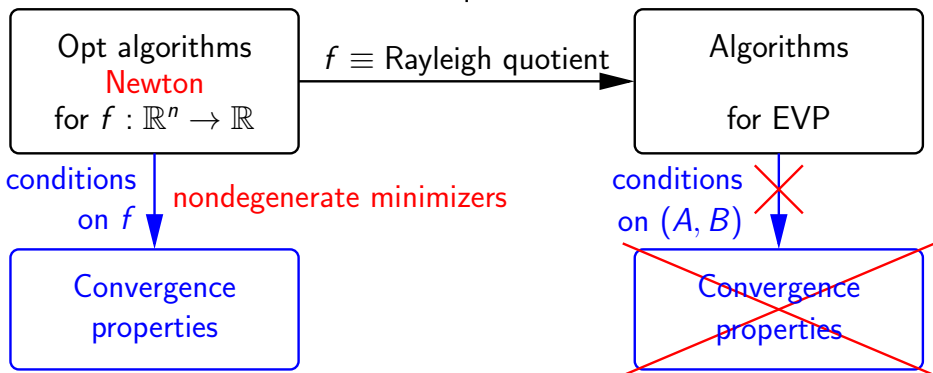
Generalized (“block”) Rayleigh quotient:

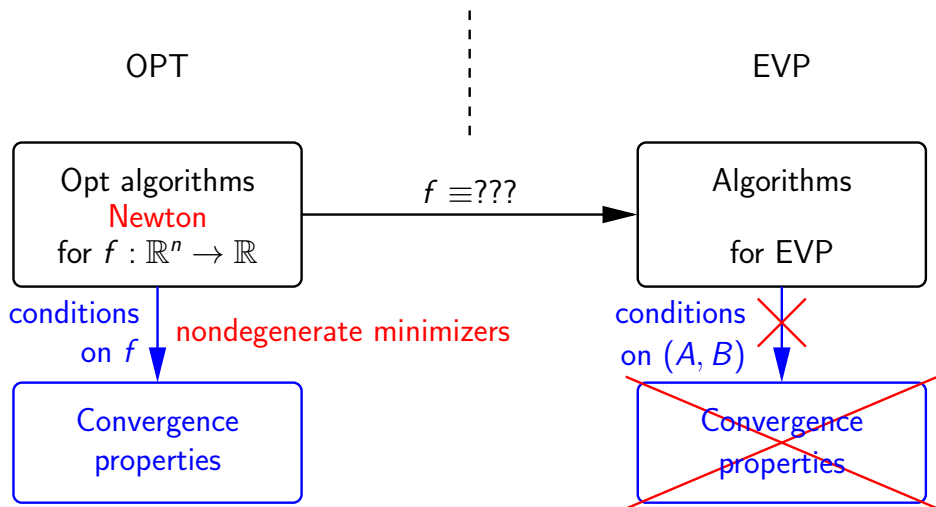
$$f : \mathbb{R}_*^{n \times p} \rightarrow \mathbb{R} : f(Y) = \text{trace} \left((Y^T B Y)^{-1} Y^T A Y \right)$$

Invariance: $f(YM) = f(Y)$ for all $M \in \mathbb{R}_*^{p \times p}$.

OPT

EVP



Remedy 1: modify f 

Remedy 1: modify f

Consider

$$P_A : \mathbb{R}^n \rightarrow \mathbb{R} : x \mapsto P_A(x) := (x^T x)^2 - 2x^T Ax.$$

Theorem

(i)

$$\min_{x \in \mathbb{R}^n} P_A(x) = -\lambda_n^2$$

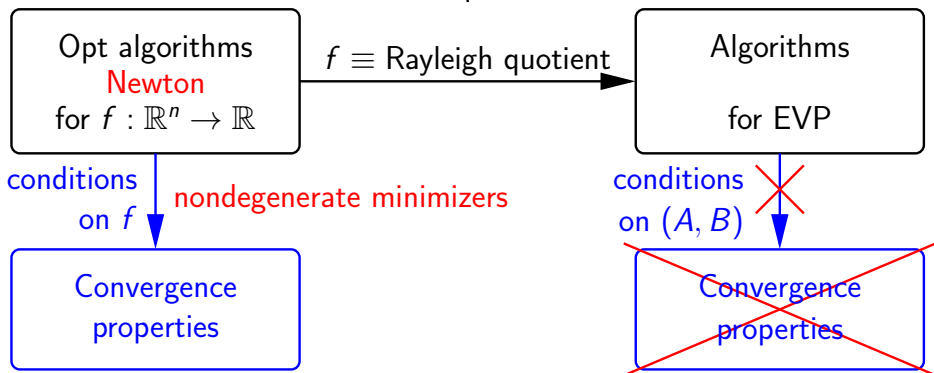
The minimum is attained at any $\sqrt{\lambda_n} v_n$, where v_n is a **unitary** eigenvector related to λ_n .

(ii) The set of critical points of P_A is $\{0\} \cup \{\sqrt{\lambda_k} v_k\}$.

References: Auchmuty (1989), Mongeau and Torki (2004).

OPT

EVP



Remedy 2: modify the search space

Instead of

$$f : \mathbb{R}_*^n \rightarrow \mathbb{R} : f(y) = \frac{y^T A y}{y^T B y},$$

minimize

$$f : \mathcal{M} \rightarrow \mathbb{R} : f(y) = \frac{y^T A y}{y^T B y},$$

where

$$\mathcal{M} = \{y \in \mathbb{R}^n : y^T B y = 1\}.$$

Stationary points of f : $\pm v_i$.

Local (and global) minimizers of f : $\pm v_1$.

Remedy 2: modify search space: block case

Instead of generalized (“block”) Rayleigh quotient:

$$f : \mathbb{R}_*^{n \times p} \rightarrow \mathbb{R} : f(Y) = \text{trace} \left((Y^T B Y)^{-1} Y^T A Y \right),$$

minimize

$$f : \text{Grass}(p, n) \rightarrow \mathbb{R} : f(\text{col}(Y)) = \text{trace} \left((Y^T B Y)^{-1} Y^T A Y \right),$$

where $\text{Grass}(p, n)$ denotes the set of all p -dimensional subspaces of \mathbb{R}^n , called the *Grassmann manifold*.

Stationary points of f : $\text{col}([v_{i_1} \ \dots \ v_{i_p}])$.

Minimizer of f : $\text{col}([v_1 \ \dots \ v_p])$.

OPT

Opt algorithms
Newton
for $f : \mathcal{M} \rightarrow \mathbb{R}$

conditions
on f

nondegenerate minimizers

Convergence
properties

$f \equiv$ Rayleigh quotient

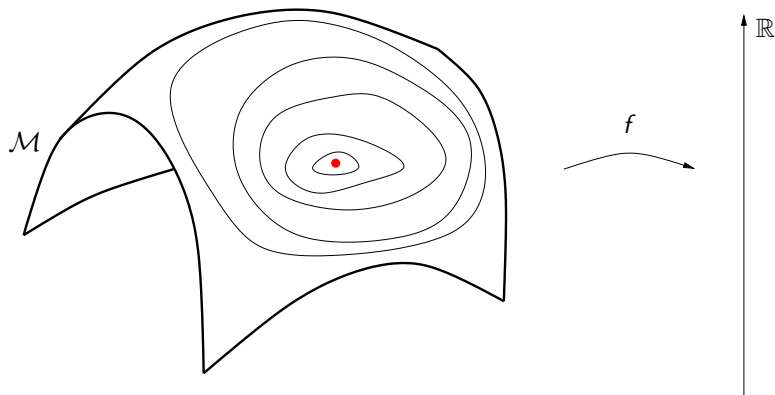
EVP

Algorithms
for EVP

conditions
on (A, B)

Convergence
properties

Smooth optimization on a manifold: big picture



Smooth optimization on a manifold: tools

	Purely Riemannian way	Pragmatic way
Search direction	Tangent vector	Tangent vector
Steepest descent dir.	$-\text{grad } f(x)$	$-\text{grad } f(x)$
Derivative of vector field	Levi-Civita connection $\overset{g}{\nabla}$	Any connection ∇
Update	Search along the geodesic tangent to the search direction	Search along any curve tangent to the search direction (prescribed by a <i>retraction</i>)
Displacement of tgt vectors	Parallel translation induced by $\overset{g}{\nabla}$	Vector Transport

OPT

Opt algorithms
Newton
for $f : \mathcal{M} \rightarrow \mathbb{R}$

conditions
on f

nondegenerate minimizers

Convergence
properties

$f \equiv$ Rayleigh quotient

EVP

Algorithms
for EVP

conditions
on (A, B)

Convergence
properties

Newton's method on abstract manifolds

Required: Riemannian manifold \mathcal{M} ; retraction R on \mathcal{M} ; affine connection ∇ on \mathcal{M} ; real-valued function f on \mathcal{M} .

Iteration $x_k \in \mathcal{M} \mapsto x_{k+1} \in \mathcal{M}$ defined by

1. Solve the Newton equation

$$\text{Hess } f(x_k)\eta_k = -\text{grad } f(x_k)$$

for the unknown $\eta_k \in T_{x_k}\mathcal{M}$, where $\text{Hess } f(x_k)\eta_k := \nabla_{\eta_k}\text{grad } f$.

2. Set

$$x_{k+1} := R_{x_k}(\eta_k).$$

OPT

Opt algorithms
Newton
for $f : \mathcal{M} \rightarrow \mathbb{R}$

conditions
on f

nondegenerate minimizers

Convergence
properties

$f \equiv$ Rayleigh quotient

EVP

Algorithms
for EVP

conditions
on (A, B)

Convergence
properties

Convergence of Newton's method on abstract manifolds

Theorem

Let $x_* \in \mathcal{M}$ be a **nongenerate critical point** of f , i.e., $\text{grad } f(x_*) = 0$ and $\text{Hess } f(x_*)$ invertible.

Then there exists a neighborhood \mathcal{U} of x_* in \mathcal{M} such that, for all $x_0 \in \mathcal{U}$, Newton's method generates an infinite sequence $(x_k)_{k=0,1,\dots}$ **converging superlinearly** (at least quadratically) to x_* .

OPT

Opt algorithms
Newton
 for $f : \mathcal{M} \rightarrow \mathbb{R}$

conditions
 on f

nondegenerate minimizers

Convergence
 properties

$f \equiv$ Rayleigh quotient

EVP

Algorithms
 for EVP

conditions
 on (A, B)

Convergence
 properties

Geometric Newton for Rayleigh quotient optimization

Iteration $x_k \in S^{n-1} \mapsto x_{k+1} \in S^{n-1}$ defined by

1. Solve the Newton equation

$$\begin{cases} P_{x_k} A P_{x_k} \eta_k - \eta_k x_k^T A x_k = -P_{x_k} A x_k, \\ x_k^T \eta_k = 0, \end{cases}$$

for the unknown $\eta_k \in \mathbb{R}^n$, where

$$P_{x_k} = (I - x_k x_k^T).$$

2. Set

$$x_{k+1} := \frac{x_k + \eta_k}{\|x_k + \eta_k\|}.$$

Geometric Newton for Rayleigh quotient optimization: block case

Iteration $\text{col}(Y_k) \in \text{Grass}(p, n) \mapsto \text{col}(Y_{k+1}) \in \text{Grass}(p, n)$ defined by

1. Solve the linear system

$$\begin{cases} P_{Y_k}^h (AZ_k - Z_k(Y_k^T Y_k)^{-1} Y_k^T A Y_k) = -P_{Y_k}^h (A Y_k) \\ Y_k^T Z_k = 0 \end{cases}$$

for the unknown $Z_k \in \mathbb{R}^{n \times p}$, where

$$P_{Y_k}^h = (I - Y_k(Y_k^T Y_k)^{-1} Y_k^T).$$

2. Set

$$Y_{k+1} = (Y_k + Z_k)N_k$$

where N_k is a nonsingular $p \times p$ matrix chosen for normalization.

OPT

Opt algorithms
Newton
for $f : \mathcal{M} \rightarrow \mathbb{R}$

conditions
on f

nondegenerate minimizers

Convergence
properties

$f \equiv$ Rayleigh quotient

EVP

Algorithms
for EVP

conditions
on (A, B)

Convergence
properties

Convergence of the EVP algorithm

Theorem

Let $Y_* \in \mathbb{R}^{n \times p}$ be such that $\text{col}(Y_*)$ is a **spectral** invariant subspace of $B^{-1}A$. Then there exists a neighborhood \mathcal{U} of $\text{col}(Y_*)$ in $\text{Grass}(p, n)$ such that, for all $Y_0 \in \mathbb{R}^{n \times p}$ with $\text{col}(Y_0) \in \mathcal{U}$, Newton's method generates an infinite sequence $(Y_k)_{k=0,1,\dots}$ such that $(\text{col}(Y_k))_{k=0,1,\dots}$ **converges superlinearly** (at least quadratically) to $\text{col}(Y_*)$ on $\text{Grass}(p, n)$.

OPT

Opt algorithms
Newton
for $f : \mathcal{M} \rightarrow \mathbb{R}$

conditions
on f

nondegenerate minimizers

Convergence
properties

$f \equiv$ Rayleigh quotient

EVP

Algorithms
for EVP

conditions
on (A, B)

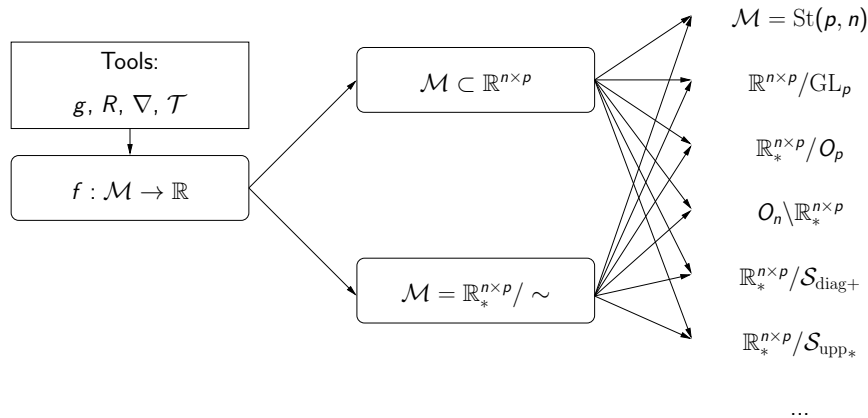
Convergence
properties

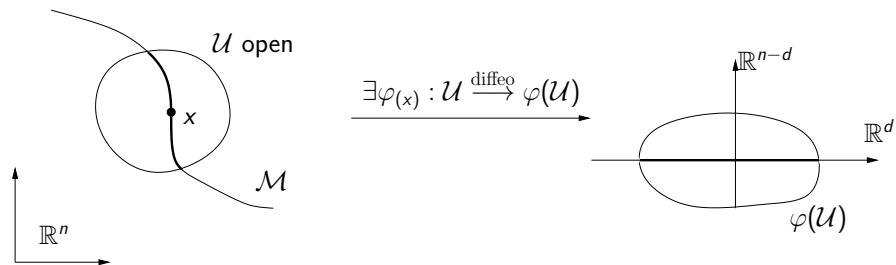
Other optimization methods

- ▶ Trust-region methods: PAA, C. G. Baker, K. A. Gallivan, *Trust-region methods on Riemannian manifolds*, Foundations of Computational Mathematics, 2007.
- ▶ “Implicit” trust-region methods: PAA, C. G. Baker, K. A. Gallivan, *An implicit trust-region method on Riemannian manifolds*, IMAJNA, 2008. The RTR solver in Sandia Lab’s Trilinos Anasazi framework (<https://trilinos.org/packages/anasazi/>) is based on this work.

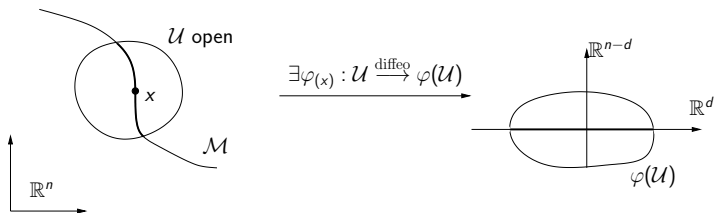
Manifolds

Manifolds, submanifolds, quotient manifolds



Submanifolds of \mathbb{R}^n 

The set $\mathcal{M} \subset \mathbb{R}^n$ is termed a *submanifold* of \mathbb{R}^n if the situation described above holds for all $x \in \mathcal{M}$.

Submanifolds of \mathbb{R}^n 

The manifold structure on \mathcal{M} is defined in a unique way as the manifold

structure generated by the atlas $\left\{ \left[\begin{array}{c} e_1^T \\ \vdots \\ e_d^T \end{array} \right] \varphi(x)|_{\mathcal{M}} : x \in \mathcal{M} \right\}$.

Back to the basics: partial derivatives in \mathbb{R}^n

Let $F : \mathbb{R}^n \rightarrow \mathbb{R}^q$.

Define $\partial_i F : \mathbb{R}^n \rightarrow \mathbb{R}^q$ by

$$\partial_i F(x) = \lim_{t \rightarrow 0} \frac{F(x + te_i) - F(x)}{t}.$$

If $\partial_i F$ is defined and continuous on \mathbb{R}^n , then F is termed *continuously differentiable*, denoted by $F \in C^1$.

Back to the basics: (Fréchet) derivative in \mathbb{R}^n

If $F \in C^1$, then

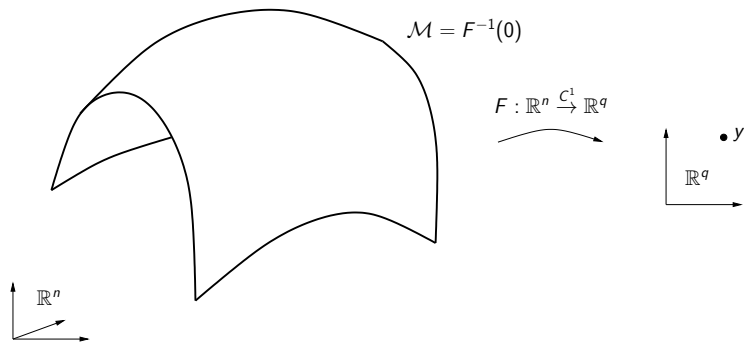
$$DF(x) : \mathbb{R}^n \xrightarrow{\text{lin}} \mathbb{R}^q : z \mapsto DF(x)[z] := \lim_{t \rightarrow 0} \frac{F(x + tz) - F(x)}{t}$$

is the *derivative* (or *differential*) of F at x .

We have $DF(x)[z] = J_F(x)z$, where the matrix

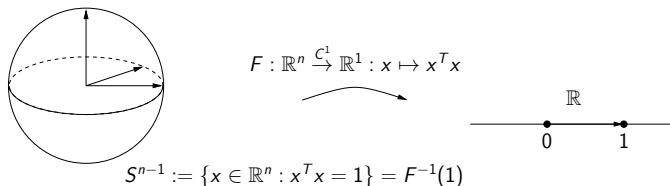
$$J_F(x) = \begin{bmatrix} \partial_1(e_1^T F)(x) & \cdots & \partial_n(e_1^T F)(x) \\ \vdots & \ddots & \vdots \\ \partial_1(e_q^T F)(x) & \cdots & \partial_n(e_q^T F)(x) \end{bmatrix}$$

is the *Jacobian matrix* of F at x .

Submanifolds of \mathbb{R}^n : sufficient condition

$y \in \mathbb{R}^q$ is a *regular value* of F if, for all $x \in F^{-1}(y)$, $DF(x)$ is an onto function (*surjection*).

Theorem (submersion theorem): If $y \in \mathbb{R}^q$ is a regular value of F , then $F^{-1}(y)$ is a submanifold of \mathbb{R}^n .

Submanifolds of \mathbb{R}^n : sufficient condition: application

The unit sphere

$$S^{n-1} := \{x \in \mathbb{R}^n : x^T x = 1\}$$

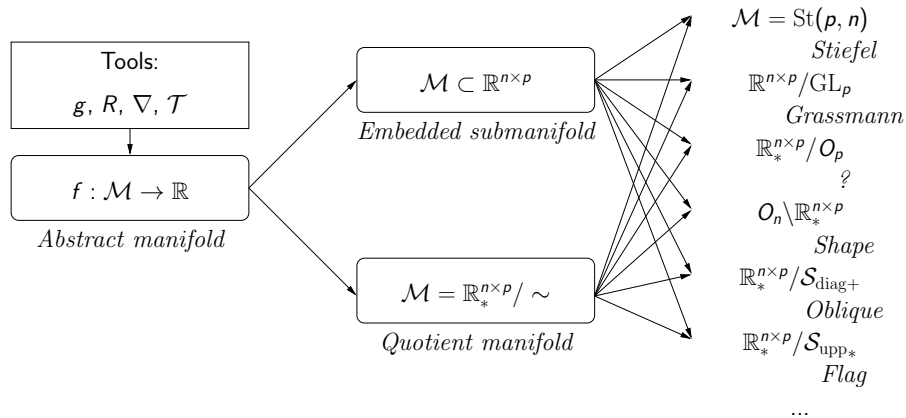
is a submanifold of \mathbb{R}^n .

Indeed, for all $x \in S^{n-1}$, we have that

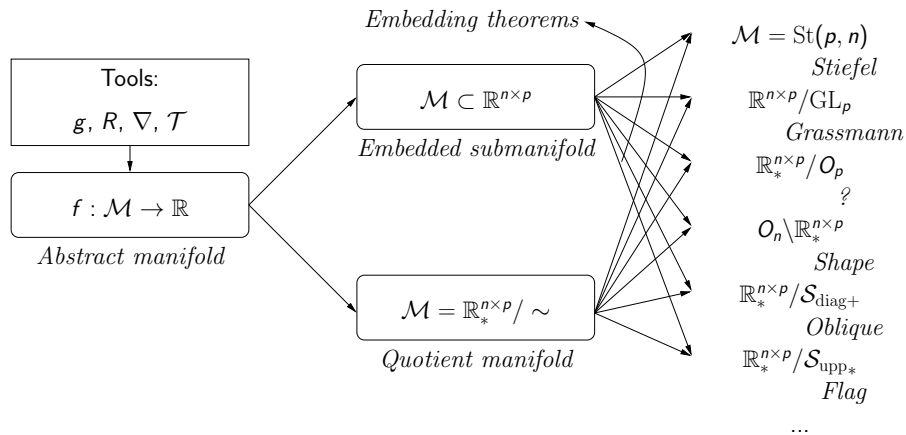
$$DF(x) : \mathbb{R}^n \rightarrow \mathbb{R} : z \mapsto DF(x)[z] = x^T z + z^T x$$

is an onto function.

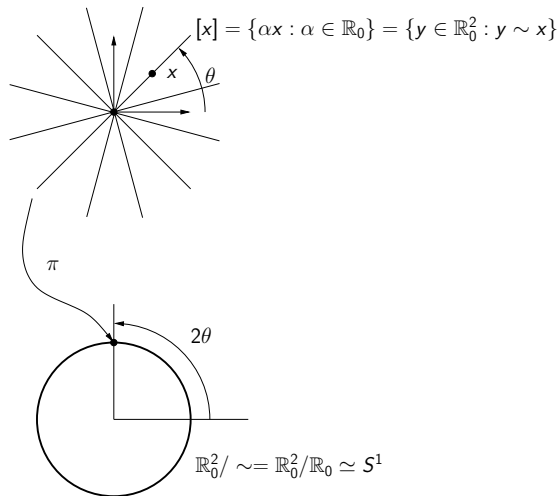
Manifolds, submanifolds, quotient manifolds



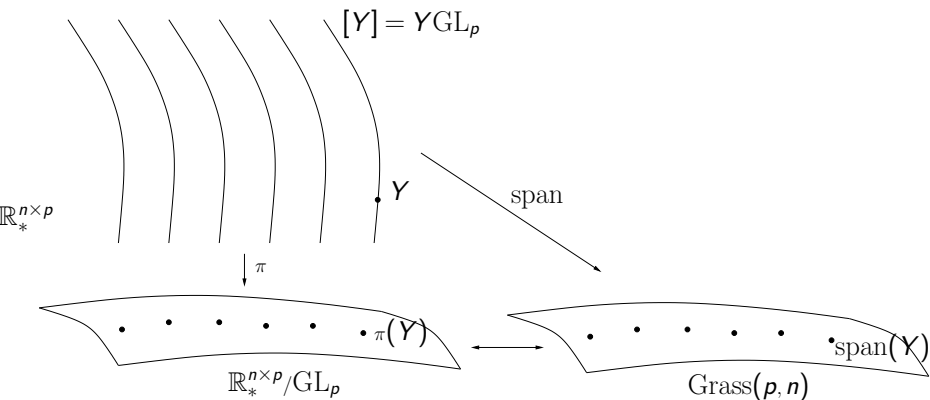
Manifolds, submanifolds, quotient manifolds

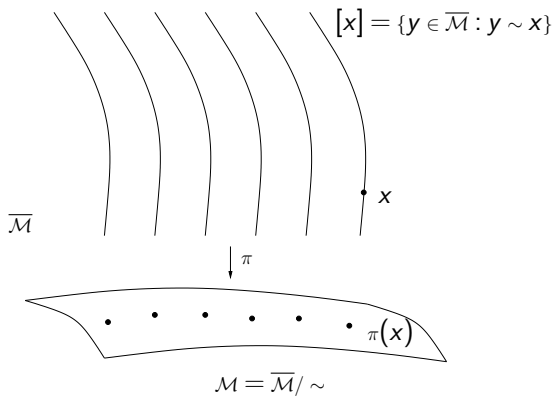


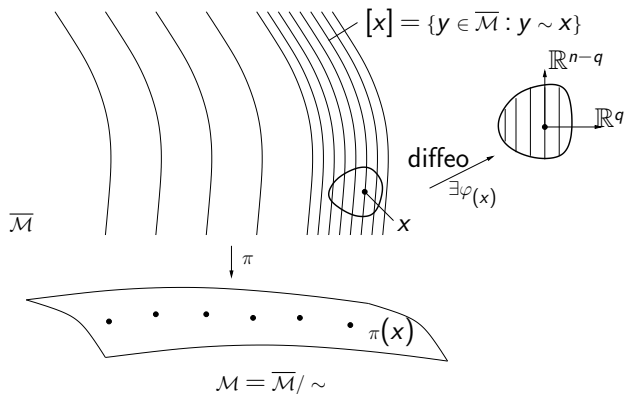
A simple quotient set: the projective space



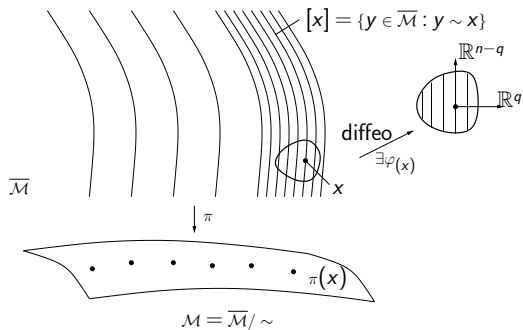
A slightly less simple quotient set: $\mathbb{R}_*^{n \times p} / \text{GL}_p$



Abstract quotient set $\overline{\mathcal{M}}/\sim$ 

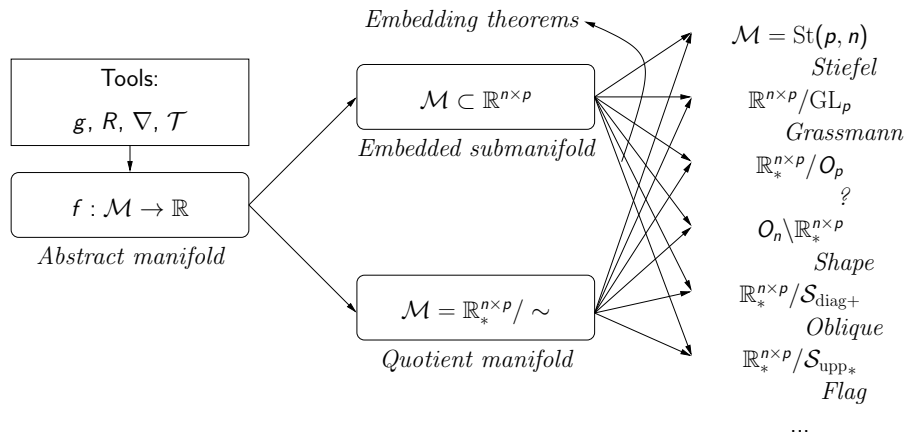
Abstract quotient manifold $\overline{\mathcal{M}}/\sim$ 

The set $\overline{\mathcal{M}}/\sim$ is termed a *quotient manifold* if the situation described above holds for all $x \in \overline{\mathcal{M}}$.

Abstract quotient manifold $\overline{\mathcal{M}}/\sim$ 

The manifold structure on $\overline{\mathcal{M}}/\sim$ is defined in a unique way as the manifold structure generated by the atlas $\left\{ \left[\begin{array}{c} e_1^T \\ \vdots \\ e_q^T \end{array} \right] \varphi(x) \circ \pi^{-1} : x \in \overline{\mathcal{M}} \right\}$.

Manifolds, submanifolds, quotient manifolds



Manifolds, and where they appear

- ▶ Stiefel manifold $\text{St}(p, n)$ and orthogonal group $O_p = \text{St}(n, n)$

$$\text{St}(p, n) = \{X \in \mathbb{R}^{n \times p} : X^T X = I_p\}$$

Applications: computer vision; principal component analysis; independent component analysis...

- ▶ Grassmann manifold $\text{Grass}(p, n)$

Set of all p -dimensional subspaces of \mathbb{R}^n

Applications: various dimension reduction problems...

- ▶ $\mathbb{R}_*^{n \times p} / O_p$

$$X \sim Y \Leftrightarrow \exists Q \in O_p : Y = XQ$$

Applications: Low-rank approximation of symmetric matrices; low-rank approximation of tensors...

Manifolds, and where they appear

- ▶ Shape manifold $O_n/\mathbb{R}_*^{n \times p}$

$$Y \sim Y' \Leftrightarrow \exists U \in O_n : Y' = UY$$

Applications: shape analysis

- ▶ Oblique manifold $\mathbb{R}_*^{n \times p}/\mathcal{S}_{\text{diag}+}$

$$\mathbb{R}_*^{n \times p}/\mathcal{S}_{\text{diag}+} \simeq \{Y \in \mathbb{R}_*^{n \times p} : \text{diag}(Y^T Y) = I_p\}$$

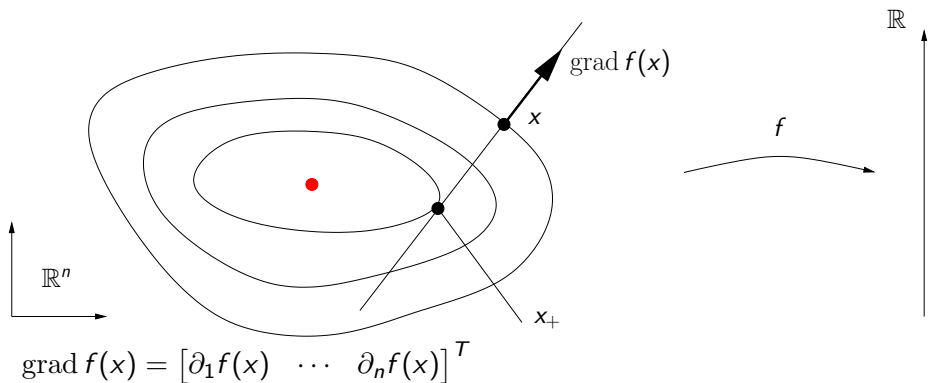
Applications: independent component analysis; factor analysis (oblique Procrustes problem)...

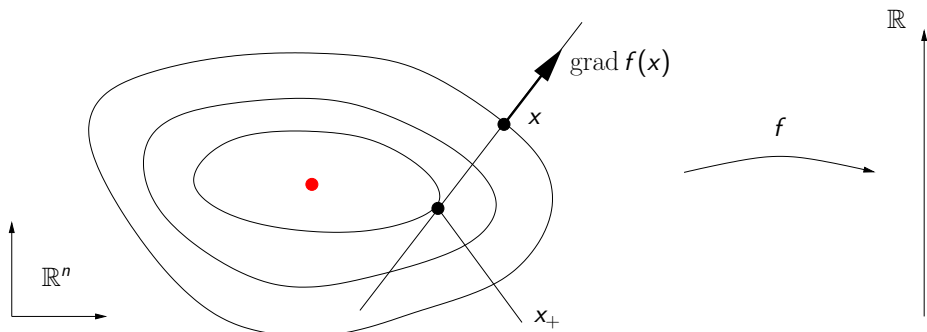
- ▶ Flag manifold $\mathbb{R}_*^{n \times p}/\mathcal{S}_{\text{upp}+}$

Elements of the flag manifold can be viewed as a p -tuple of linear subspaces $(\mathcal{V}_1, \dots, \mathcal{V}_p)$ such that $\dim(\mathcal{V}_i) = i$ and $\mathcal{V}_i \subset \mathcal{V}_{i+1}$.

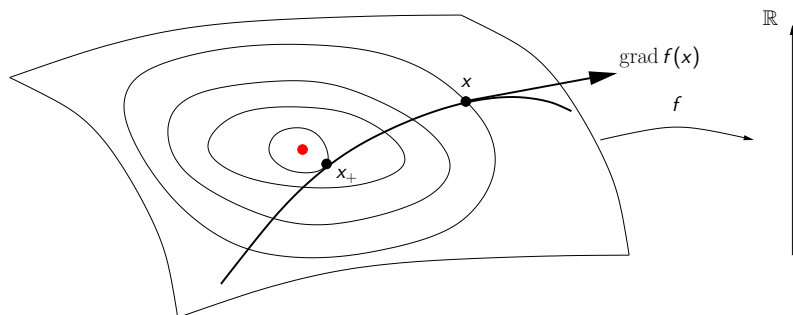
Applications: analysis of QR algorithm...

Steepest-descent methods on manifolds

Steepest-descent in \mathbb{R}^n 

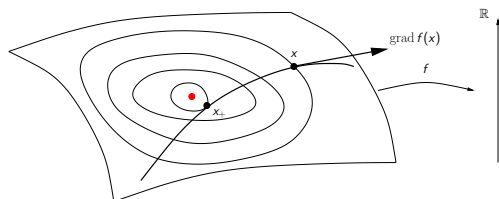
Steepest-descent: from \mathbb{R}^n to manifolds

	\mathbb{R}^n	Manifold
Search direction	Vector at x	Tangent vector at x
Steepest-desc. dir.	$-\text{grad } f(x)$	$-\text{grad } f(x)$
Curve	$\gamma : t \mapsto x - t \text{grad } f(x)$	γ s.t. $\gamma(0) = x$ and $\dot{\gamma}(0) = -\text{grad } f(x)$

Steepest-descent: from \mathbb{R}^n to manifolds

	\mathbb{R}^n	Manifold
Search direction	Vector at x	Tangent vector at x
Steepest-desc. dir.	$-\text{grad } f(x)$	$-\text{grad } f(x)$
Curve	$\gamma : t \mapsto x - t \text{grad } f(x)$	γ s.t. $\gamma(0) = x$ and $\dot{\gamma}(0) = -\text{grad } f(x)$

Update directions: tangent vectors



Let γ be a curve in the manifold \mathcal{M} with $\gamma(0) = x$.

For an abstract manifold, the definition $\dot{\gamma}(0) = \frac{d\gamma}{dt}(0) = \lim_{t \rightarrow 0} \frac{\gamma(t) - \gamma(0)}{t}$ is meaningless.

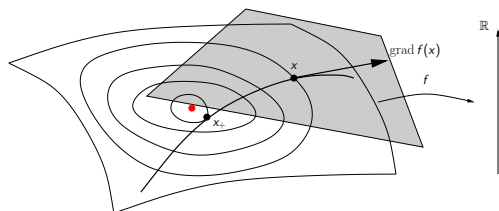
Instead, define: $Df(x)[\dot{\gamma}(0)] := \left. \frac{d}{dt} f(\gamma(t)) \right|_{t=0}$

If $\mathcal{M} \subset \mathbb{R}^n$ and $f = \bar{f}|_{\mathcal{M}}$, then

$$Df(x)[\dot{\gamma}(0)] = D\bar{f}(x) \left[\frac{d\gamma}{dt}(0) \right].$$

The application $\dot{\gamma}(0) : f \mapsto Df(x)[\dot{\gamma}(0)]$ is a *tangent vector* at x .

Update directions: tangent spaces



The set

$$T_x \mathcal{M} = \{ \dot{\gamma}(0) : \gamma \text{ curve in } \mathcal{M} \text{ through } x \text{ at } t = 0 \}$$

is the *tangent space* to \mathcal{M} at x .

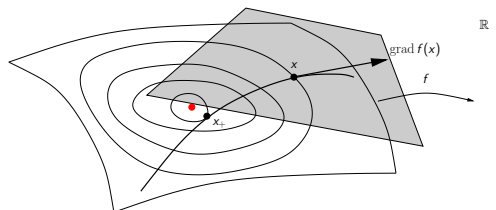
With the definition

$$\alpha \dot{\gamma}_1(0) + \beta \dot{\gamma}_2(0) : f \mapsto \alpha Df(x)[\dot{\gamma}_1(0)] + \beta Df(x)[\dot{\gamma}_2(0)],$$

the tangent space $T_x \mathcal{M}$ becomes a *linear space*.

The *tangent bundle* $T\mathcal{M}$ is the set of all tangent vectors to \mathcal{M} .

Tangent vectors: submanifolds of Euclidean spaces

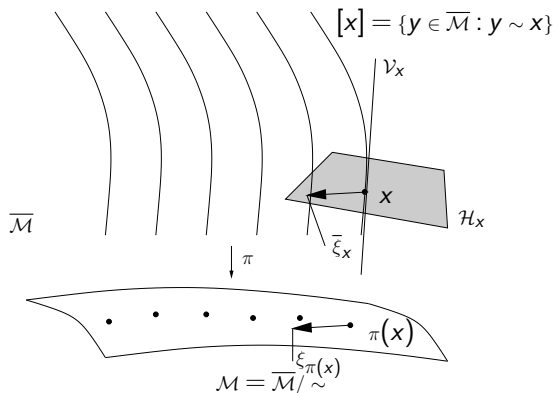


If \mathcal{M} is a submanifold of \mathbb{R}^n and $f = \bar{f}|_{\mathcal{M}}$, then

$$Df(x)[\dot{\gamma}(0)] = D\bar{f}(x) \left[\frac{d\gamma}{dt}(0) \right].$$

Proof: The left-hand side is equal to $\left. \frac{d}{dt} f(\gamma(t)) \right|_{t=0}$. This is equal to $\left. \frac{d}{dt} \bar{f}(\gamma(t)) \right|_{t=0}$ because $\gamma(t) \in \mathcal{M}$ for all t . The classical chain rule yields the right-hand side.

Tangent vectors: quotient manifolds



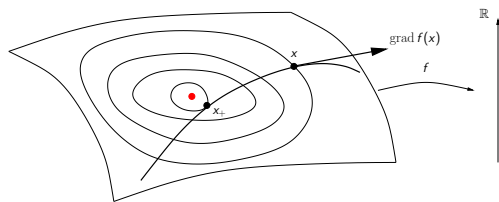
Let $\overline{\mathcal{M}}/\sim$ be a quotient manifold. Then $[x]$ is a submanifold of $\overline{\mathcal{M}}$. The tangent space $T_x[x]$ is the *vertical space* \mathcal{V}_x . A *horizontal space* is a subspace of $T_x \overline{\mathcal{M}}$ complementary to \mathcal{V}_x .

Let $\xi_{\pi(x)}$ be a tangent vector to $\overline{\mathcal{M}}/\sim$ at $\pi(x)$.

Theorem: In \mathcal{H}_x there is one and only one $\bar{\xi}_x$ such that

$$D\pi(x)[\bar{\xi}_x] = \xi_{\pi(x)}.$$

Steepest-descent: norm of tangent vectors



The steepest ascent direction is along

$$\arg \max_{\substack{\xi \in T_x \mathcal{M} \\ \|\xi\|=1}} Df(x)[\xi].$$

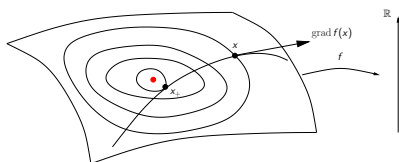
To this end, we need a norm on $T_x \mathcal{M}$.

For all $x \in \mathcal{M}$, let g_x denote an inner product in $T_x \mathcal{M}$, and define

$$\|\xi_x\| := \sqrt{g_x(\xi_x, \xi_x)}.$$

When g_x “smoothly” depends on x , we say that (\mathcal{M}, g) is a *Riemannian manifold*.

Steepest-descent: gradient



There is a unique $\text{grad } f(x)$, called the *gradient* of f at x , such that

$$\begin{cases} \text{grad } f(x) \in T_x \mathcal{M} \\ \mathbf{g}_x(\text{grad } f(x), \xi_x) = Df(x)[\xi_x], \quad \forall \xi_x \in T_x \mathcal{M}. \end{cases}$$

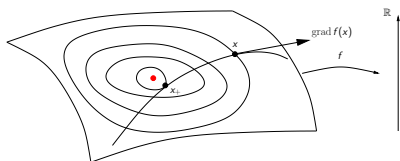
We have

$$\frac{\text{grad } f(x)}{\|\text{grad } f(x)\|} = \arg \max_{\substack{\xi \in T_x \mathcal{M} \\ \|\xi\|=1}} Df(x)[\xi]$$

and

$$\|\text{grad } f(x)\| = Df(x) \left[\frac{\text{grad } f(x)}{\|\text{grad } f(x)\|} \right].$$

Steepest-descent: Riemannian submanifolds



Let $(\overline{\mathcal{M}}, \overline{g})$ be a Riemannian manifold and \mathcal{M} be a submanifold of $\overline{\mathcal{M}}$. Then

$$g_x(\xi_x, \zeta_x) := \overline{g}_x(\xi_x, \eta_x), \quad \forall \xi_x, \zeta_x \in T_x \mathcal{M}$$

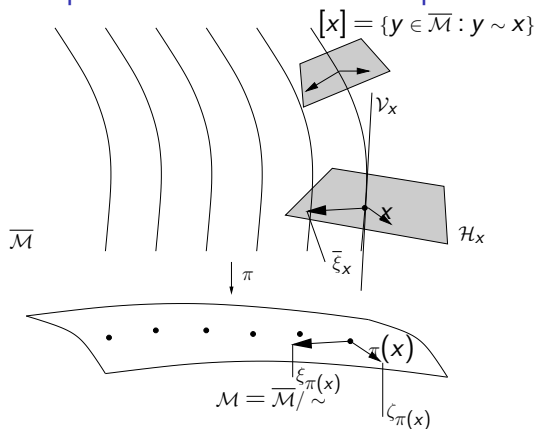
defines a Riemannian metric g on \mathcal{M} . With this Riemannian metric, \mathcal{M} is a *Riemannian submanifold* of $\overline{\mathcal{M}}$.

Every $z \in T_x \overline{\mathcal{M}}$ admits a decomposition $z = \underbrace{P_x z}_{\in T_x \mathcal{M}} + \underbrace{P_x^\perp z}_{\in T_x^\perp \mathcal{M}}$.

If $\overline{f} : \overline{\mathcal{M}} \rightarrow \mathbb{R}$ and $f = \overline{f}|_{\mathcal{M}}$, then

$$\text{grad } f(x) = P_x \text{grad } \overline{f}(x).$$

Steepest-descent: Riemannian quotient manifolds

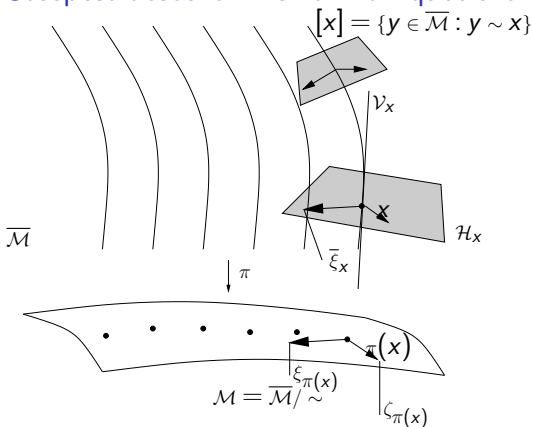


Let \overline{g} be a Riemannian metric on $\overline{\mathcal{M}}$.

Suppose that, for all $\xi_{\pi(x)}$ and $\zeta_{\pi(x)}$ in $T_{\pi(x)}\overline{\mathcal{M}} / \sim$, and all $\tilde{x} \in \pi^{-1}(\pi(x))$, we have

$$\overline{g}_{\tilde{x}}(\overline{\xi}_{\tilde{x}}, \overline{\zeta}_{\tilde{x}}) = \overline{g}_x(\overline{\xi}_x, \overline{\zeta}_x).$$

Steepest-descent: Riemannian quotient manifolds



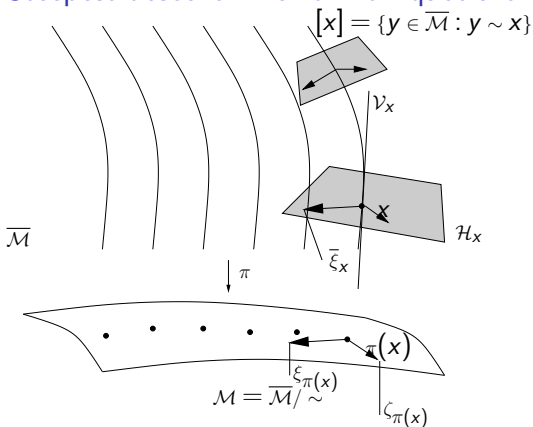
Then

$$\mathbf{g}_{\pi(x)}(\xi_{\pi(x)}, \zeta_{\pi(x)}) := \bar{\mathbf{g}}_x(\bar{\xi}_x, \bar{\zeta}_x).$$

defines a Riemannian metric on $\overline{\mathcal{M}} / \sim$.

If moreover \mathcal{H}_x is the orthogonal complement of ν_x in the sense of $\bar{\mathbf{g}}$, then π is termed a *Riemannian submersion*, and we say that $\overline{\mathcal{M}} / \sim$ is a *Riemannian quotient manifold*.

Steepest-descent: Riemannian quotient manifolds



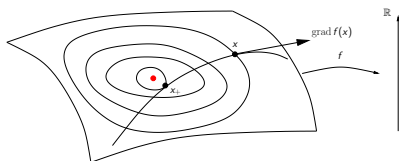
Let $f : \overline{\mathcal{M}} / \sim \rightarrow \mathbb{R}$. Let $P_x^{h, \overline{g}}$ denote the orthogonal projection onto \mathcal{H}_x .

$$\overline{\text{grad}} f_x = P_x^{h, \overline{g}} \text{grad}(f \circ \pi)(x).$$

If \mathcal{H}_x is the orthogonal complement of \mathcal{V}_x in the sense of \overline{g} (π is a *Riemannian submersion*), then $\text{grad}(f \circ \pi)(x)$ is already in \mathcal{H}_x , and thus

$$\overline{\text{grad}} f_x = \text{grad}(f \circ \pi)(x).$$

Steepest-descent: choosing the search curve



It remains to choose a curve γ through x at $t = 0$ such that

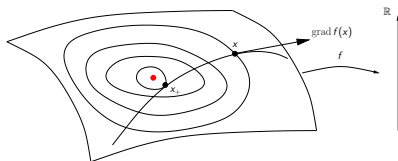
$$\dot{\gamma}(0) = -\text{grad } f(x).$$

Let $R : T\mathcal{M} \rightarrow \mathcal{M}$ be a *retraction* on \mathcal{M} , that is

1. $R(0_x) = x$, where 0_x denotes the origin of $T_x\mathcal{M}$;
2. $\frac{d}{dt}R(t\xi_x) = \xi_x$.

Then choose $\gamma : t \mapsto R(-t\text{grad } f(x))$.

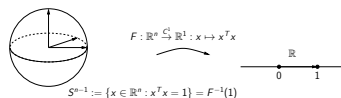
Steepest-descent: line-search procedure



Find t such that $f(\gamma(t))$ is “sufficiently smaller” than $f(\gamma(0))$. Since $t \mapsto f(\gamma(t))$ is just a function from \mathbb{R} to \mathbb{R} , we can use the step selection techniques that are available for classical line-search methods.

For example: exact minimization, Armijo backtracking,...

Steepest-descent: Rayleigh quotient on unit sphere



Let the manifold be the unit sphere

$$S^{n-1} = \{x \in \mathbb{R}^n : x^T x = 1\} = F^{-1}(1),$$

where $F: \mathbb{R}^n \rightarrow \mathbb{R}: x \mapsto x^T x$.

Let $A = A^T \in \mathbb{R}^{n \times n}$ and let the cost function be the Rayleigh quotient

$$f: S^{n-1} \rightarrow \mathbb{R}: x \mapsto x^T A x.$$

The **tangent space** to S^{n-1} at x is

$$T_x S^{n-1} = \ker(DF(x)) = \{z \in \mathbb{R}^n : x^T z = 0\}.$$

Derivation formulas

If F is linear, then

$$DF(x)[z] = F(z).$$

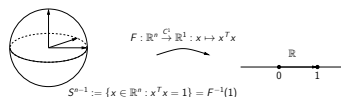
Chain rule: If $\text{range}(F) \subseteq \text{dom}(G)$, then

$$D(G \circ F)(x)[z] = DG(F(x))[DF(x)[z]].$$

Product rule: If the ranges of F and G are in matrix spaces of compatible dimension, then

$$D(FG)(x)[z] = DF(x)[z]G(x) + F(x)DG(x)[z].$$

Steepest-descent: Rayleigh quotient on unit sphere



Rayleigh quotient:

$$f: S^{n-1} \rightarrow \mathbb{R}: x \mapsto x^T A x.$$

The tangent space to S^{n-1} at x is

$$T_x S^{n-1} = \ker(DF(x)) = \{z \in \mathbb{R}^n : x^T z = 0\}.$$

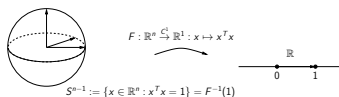
Product rule:

$$D(FG)(x)[z] = DF(x)[z]G(x) + F(x)DG(x)[z].$$

Differential of f at $x \in S^{n-1}$:

$$Df(x)[z] = x^T A z + z^T A x = 2z^T A x, \quad z \in T_x S^{n-1}.$$

Steepest-descent: Rayleigh quotient on unit sphere



“Natural” Riemannian metric on S^{n-1} :

$$g_x(z_1, z_2) = z_1^T z_2, \quad z_1, z_2 \in T_x S^{n-1}.$$

Differential of f at $x \in S^{n-1}$:

$$Df(x)[z] = 2z^T Ax = 2g_x(z, Ax), \quad z \in T_x S^{n-1}.$$

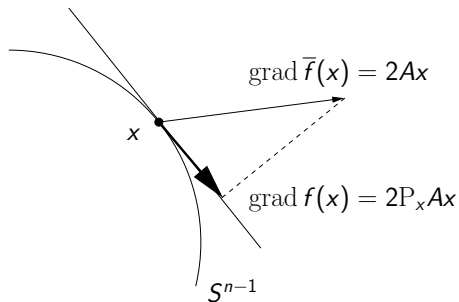
Gradient:

$$\text{grad } f(x) = 2P_x Ax = 2(I - xx^T)Ax.$$

Check:

$$\begin{cases} \text{grad } f(x) \in T_x S^{n-1} \\ Df(x)[z] = g_x(\text{grad } f(x), z), \quad \forall z \in T_x S^{n-1}. \end{cases}$$

Steepest-descent: Rayleigh quotient on unit sphere



$$f : S^{n-1} \rightarrow \mathbb{R} : x \mapsto x^T Ax$$

$$\bar{f} : \mathbb{R}^n \rightarrow \mathbb{R} : x \mapsto x^T Ax$$

$$\text{grad } \bar{f}(x) = 2Ax$$

$$\text{grad } f(x) = 2P_x Ax = 2(I - xx^T)Ax.$$

Newton's method on manifolds

Newton in \mathbb{R}^n

Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$.

Recall $\text{grad } f(x) = [\partial_1 f(x) \ \cdots \ \partial_n f(x)]^T$.

Newton's iteration:

1. Solve, for the unknown $z \in \mathbb{R}^n$,

$$D(\text{grad } f)(x)[z] = -\text{grad } f(x).$$

2. Set

$$x_+ = x + z.$$

Newton in \mathbb{R}^n : how it may fail

Let $f : \mathbb{R}_0^n \rightarrow \mathbb{R} : x \mapsto \frac{x^T A x}{x^T x}$.

Newton's iteration:

1. Solve, for the unknown $z \in \mathbb{R}^n$,

$$D(\text{grad } f)(x)[z] = -\text{grad } f(x).$$

2. Set

$$x_+ = x + z.$$

Proposition: For all x such that $f(x)$ is not an eigenvalue of A , we have

$$x_+ = 2x.$$

Newton: how to make it work for RQ

Let $f : S^{n-1} \rightarrow \mathbb{R} : x \mapsto \frac{x^T A x}{x^T x}$.

Newton's iteration:

1. Solve, for the unknown $z \in \mathbb{R}^n \rightsquigarrow \eta_x \in T_x S^{n-1}$

$$D(\text{grad } f)(x)[z] = -\text{grad } f(x) \rightsquigarrow \boxed{?}(\text{grad } f)(x)[\eta_x] = -\text{grad } f(x)$$

2. Set

$$x_+ = x + z \rightsquigarrow x_+ = R(\eta_x)$$

Newton's equation on an abstract manifold

Let \mathcal{M} be a manifold and let $f : \mathcal{M} \rightarrow \mathbb{R}$ be a cost function.
 The mapping $x \in \mathcal{M} \mapsto \text{grad } f(x) \in T_x\mathcal{M}$ is a *vector field*.

$$D(\text{grad } f)(x)[z] = -\text{grad } f(x) \rightsquigarrow \boxed{?}(\text{grad } f)(x)[\eta_x] = -\text{grad } f(x)$$

The new object has to be such that

- ▶ In \mathbb{R}^n , $\boxed{?}$ reduces to the classical derivative
- ▶ $\boxed{?}(\text{grad } f)(x)[\eta_x]$ belongs to $T_x\mathcal{M}$
- ▶ $\boxed{?}$ has the same linearity properties and multiplication rule as the classical derivative.

Newton's equation on an abstract manifold

Let \mathcal{M} be a manifold and let $f : \mathcal{M} \rightarrow \mathbb{R}$ be a cost function. The mapping $x \in \mathcal{M} \mapsto \text{grad } f(x) \in T_x\mathcal{M}$ is a *vector field*.

$$D(\text{grad } f)(x)[z] = -\text{grad } f(x) \rightsquigarrow \boxed{?}(\text{grad } f)(x)[\eta_x] = -\text{grad } f(x)$$

The new object has to be such that

- ▶ In \mathbb{R}^n , $\boxed{?}$ reduces to the classical derivative
- ▶ $\boxed{?}(\text{grad } f)(x)[\eta_x]$ belongs to $T_x\mathcal{M}$
- ▶ $\boxed{?}$ has the same linearity properties and multiplication rule as the classical derivative.

Differential geometry offers a concept that matches these conditions: the concept of an *affine connection*.

Newton: affine connections

Let $\mathfrak{X}(\mathcal{M})$ denote the set of smooth vector fields on \mathcal{M} and $\mathfrak{F}(\mathcal{M})$ the set of real-valued functions on \mathcal{M} .

An *affine connection* ∇ on a manifold \mathcal{M} is a mapping

$$\nabla : \mathfrak{X}(\mathcal{M}) \times \mathfrak{X}(\mathcal{M}) \rightarrow \mathfrak{X}(\mathcal{M}),$$

which is denoted by $(\eta, \xi) \xrightarrow{\nabla} \nabla_{\eta}\xi$ and satisfies the following properties:

- i) $\mathfrak{F}(\mathcal{M})$ -linearity in η : $\nabla_{f\eta+g\chi}\xi = f\nabla_{\eta}\xi + g\nabla_{\chi}\xi,$
- ii) \mathbb{R} -linearity in ξ : $\nabla_{\eta}(a\xi + b\zeta) = a\nabla_{\eta}\xi + b\nabla_{\eta}\zeta,$
- iii) Product rule (Leibniz' law): $\nabla_{\eta}(f\xi) = (\eta f)\xi + f\nabla_{\eta}\xi,$

in which $\eta, \chi, \xi, \zeta \in \mathfrak{X}(\mathcal{M})$, $f, g \in \mathfrak{F}(\mathcal{M})$, and $a, b \in \mathbb{R}$.

Newton's method on abstract manifolds

Cost function: $f : \mathbb{R}^n \rightarrow \mathbb{R} \rightsquigarrow f : \mathcal{M} \rightarrow \mathbb{R}$.

Newton's iteration:

1. Solve, for the unknown $z \in \mathbb{R}^n \rightsquigarrow \eta_x \in T_x \mathcal{M}$

$$D(\text{grad } f)(x)[z] = -\text{grad } f(x) \rightsquigarrow \nabla(\text{grad } f)(x)[\eta_x] = -\text{grad } f(x)$$

2. Set

$$x_+ = x + z \rightsquigarrow x_+ = R(\eta_x)$$

In the algorithm above, ∇ is an affine connection on \mathcal{M} and R is a retraction on \mathcal{M} .

Newton's method on S^{n-1}

If \mathcal{M} is a Riemannian submanifold of \mathbb{R}^n , then ∇ defined by

$$\nabla_{\eta_x} \xi = P_x D\xi(x)[\eta_x], \quad \eta_x \in T_x \mathcal{M}, \quad \xi \in \mathfrak{X}(\mathcal{M})$$

is a particular affine connection, called *Riemannian connection*.

For the unit sphere S^{n-1} , this yields

$$\nabla_{\eta_x} \xi = (I - xx^T)D\xi(x)[\eta_x], \quad x^T \eta_x = 0.$$

Newton's method for Rayleigh quotient on S^{n-1}

$$\text{Let } f : \begin{cases} \mathbb{R}^n \\ \mathcal{M} \\ S^{n-1} \end{cases} \rightarrow \mathbb{R} : x \mapsto \begin{cases} f(x) \\ f(x) \\ \frac{x^T A x}{x^T x} \end{cases} .$$

Newton's iteration:

$$1. \text{ Solve, for the unknown } z \in \mathbb{R}^n \rightsquigarrow \eta_x \in T_x \mathcal{M} \rightsquigarrow x^T \eta_x = 0$$

$$\begin{aligned} D(\text{grad } f)(x)[z] &= -\text{grad } f(x) \\ &\rightsquigarrow \nabla(\text{grad } f)(x)[\eta_x] = -\text{grad } f(x) \\ &\rightsquigarrow (I - xx^T)(A - f(x)I)\eta_x = -(I - xx^T)Ax \end{aligned}$$

2. Set

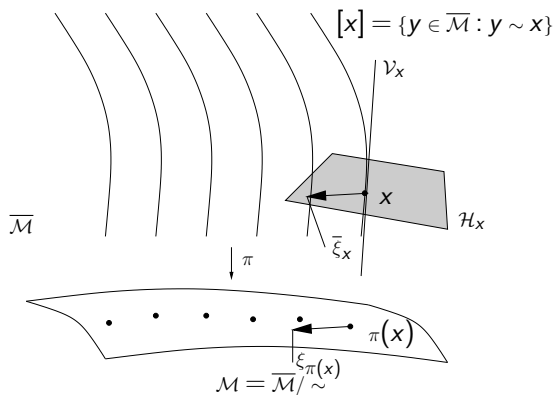
$$x_+ = x + z \rightsquigarrow x_+ = R(\eta_x) \rightsquigarrow x_+ = \frac{x + \eta_x}{\|x + \eta_x\|}$$

Newton for RQ on S^{n-1} : a closer look

$$\begin{aligned}(I - xx^T)(A - f(x)I)\eta_x &= -(I - xx^T)Ax \\ \Rightarrow (I - xx^T)(A - f(x)I)(x + \eta_x) &= 0 \\ \Rightarrow (A - f(x)I)(x + \eta_x) &= \alpha x\end{aligned}$$

Therefore, x_+ is collinear with $(A - f(x)I)^{-1}x$, which is the vector computed by the Rayleigh quotient iteration.

Newton method on quotient manifolds

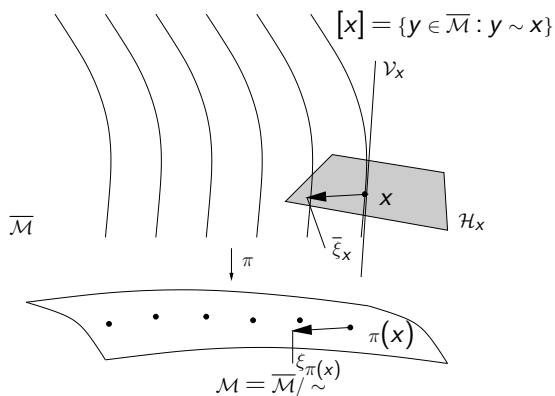


Affine connection: choose ∇ defined by

$$\overline{\nabla}_{\eta} \overline{\xi}_x = P_x^h \overline{\nabla}_{\overline{\eta}_x} \overline{\xi},$$

provided that this really defines a horizontal lift. This requires special choices of $\overline{\nabla}$.

Newton method on quotient manifolds



If $\pi : \overline{\mathcal{M}} \rightarrow \overline{\mathcal{M}} / \sim$ is a Riemannian submersion, then the Riemannian connection on $\overline{\mathcal{M}} / \sim$ is given by

$$\overline{\nabla}_{\eta} \bar{\xi}_x = P_x^h \overline{\nabla}_{\bar{\eta}_x} \bar{\xi},$$

where $\overline{\nabla}$ denotes the Riemannian connection on $\overline{\mathcal{M}}$.

A detailed exercise

Newton's method for the Rayleigh
quotient on the Grassmann
manifold

Manifold: Grassmann

The manifold is the Grassmann manifold of p -planes in \mathbb{R}^n :

$$\text{Grass}(p, n) \simeq \text{ST}(p, n)/\text{GL}_p.$$

The one-to-one correspondence is

$$\text{Grass}(p, n) \ni \mathcal{Y} \leftrightarrow Y \text{GL}_p \in \text{ST}(p, n)/\text{GL}_p$$

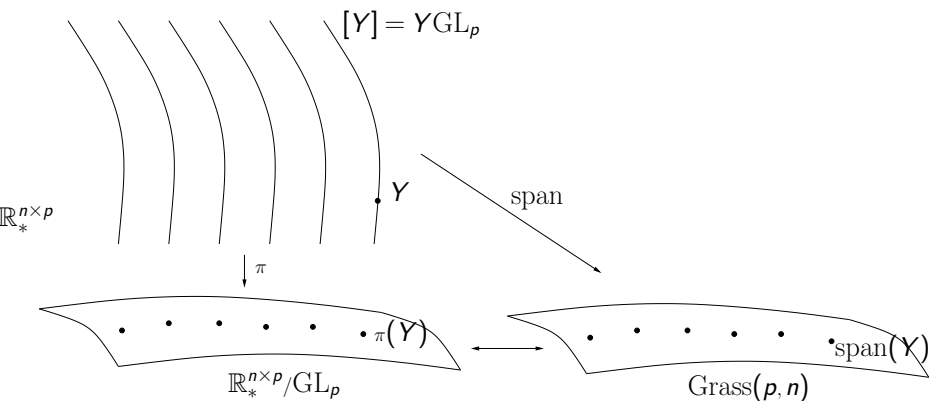
such that \mathcal{Y} is the column space of Y .

The quotient map

$$\pi : \text{ST}(p, n) \rightarrow \text{Grass}(p, n)$$

is the “column space” or “span” operation.

Grassmann and its quotient representation



Total space: the noncompact Stiefel manifold

The total space of the quotient is

$$\text{ST}(\rho, n) = \{Y \in \mathbb{R}^{n \times \rho} : \text{rank}(Y) = \rho\}.$$

This is an open submanifold of the Euclidean space $\mathbb{R}^{n \times \rho}$.

Tangent spaces: $T_Y \text{ST}(\rho, n) \simeq \mathbb{R}^{n \times \rho}$.

Riemannian metric on the total space

Define a Riemannian metric \bar{g} on $ST(p, n)$ by

$$\bar{g}_Y(Z_1, Z_2) = \text{trace} \left((Y^T Y)^{-1} Z_1^T Z_2 \right).$$

This is not the canonical Riemannian metric, but it will allow us to turn the quotient map $\pi : ST(p, n) \rightarrow \text{Grass}(p, n)$ into a Riemannian submersion.

Vertical and horizontal spaces

The vertical spaces are the tangent spaces to the equivalence classes:

$$\mathcal{V}_Y := T_Y(YGL_p) = Y T_Y GL_p = Y \mathbb{R}^{p \times p}.$$

Choice of horizontal space:

$$\begin{aligned} \mathcal{H}_Y &:= (\mathcal{V}_Y)^\perp \\ &= \{Z \in T_Y ST(p, n) : \bar{g}_Y(Z, V) = 0, \forall V \in \mathcal{V}_Y\} \\ &= \{Z \in \mathbb{R}^{n \times p} : Y^T Z = 0\}. \end{aligned}$$

Horizontal projection:

$$P_Y^h = (I - Y(Y^T Y)^{-1} Y^T).$$

Compatibility equation for horizontal lifts

Given $\xi \in T_\pi(Y)\text{Grass}(p, n)$, we have

$$\bar{\xi}_{YM} = \bar{\xi}_Y M.$$

To see this, observe that $\bar{\xi}_Y M$ is in \mathcal{H}_{YM} ; moreover, since $YM + t\bar{\xi}_Y M$ and $Y + t\bar{\xi}_Y$ have the same column space for all t , one has

$$D\pi(YM)[\bar{\xi}_Y M] = D\pi(Y)[\bar{\xi}_Y] = \xi_{\pi(Y)}.$$

Thus $\bar{\xi}_Y M$ satisfies the conditions to be $\bar{\xi}_{YM}$.

Riemannian metric on the quotient

On $\text{Grass}(p, n) \simeq \text{ST}(p, n)/\text{GL}_p$, define the Riemannian metric g by

$$g_{\pi(Y)}(\xi_{\pi(Y)}, \zeta_{\pi(Y)}) = \bar{g}_Y(\bar{\xi}_Y, \bar{\zeta}_Y).$$

This is well defined, because for all $\tilde{Y} \in \pi^{-1}(\pi(Y)) = Y\text{GL}_p$, we have $\tilde{Y} = YM$ for some invertible M , and

$$\bar{g}_{YM}(\bar{\xi}_{YM}, \bar{\zeta}_{YM}) = \bar{g}_Y(\bar{\xi}_Y, \bar{\zeta}_Y).$$

This definition of g turns

$$\pi : (\text{ST}(p, n), \bar{g}) \rightarrow (\text{Grass}(p, n), g)$$

into a Riemannian submersion.

Cost function: Rayleigh quotient

Consider the cost function

$$f : \text{Grass}(p, n) \rightarrow \mathbb{R} : \text{span}(Y) \mapsto \text{trace} \left((Y^T Y)^{-1} Y^T A Y \right).$$

This is the *projection* of

$$\bar{f} : \text{ST}(p, n) \rightarrow \mathbb{R} : Y \mapsto \text{trace} \left((Y^T Y)^{-1} Y^T A Y \right).$$

That is, $\bar{f} = f \circ \pi$.

Gradient of the cost function

For all $Z \in \mathbb{R}^{n \times p}$,

$$D\bar{f}(Y)[Z] = 2 \operatorname{trace} \left((Y^T Y)^{-1} Z^T (AY - Y(Y^T Y)^{-1} Y^T AY) \right).$$

Hence

$$\operatorname{grad} \bar{f}(Y) = 2 \left(AY - Y(Y^T Y)^{-1} Y^T AY \right),$$

and

$$\overline{\operatorname{grad} f}_Y = 2 \left(AY - Y(Y^T Y)^{-1} Y^T AY \right).$$

Riemannian connection

The quotient map is a Riemannian submersion. Therefore

$$\overline{\nabla_{\eta} \xi} = P_Y^h (\overline{\nabla_{\bar{\eta}_Y} \bar{\xi}})$$

It turns out that

$$\overline{\nabla_{\eta} \xi} = P_Y^h (D\bar{\xi}(Y) [\bar{\eta}_Y]).$$

(This is because the Riemannian metric \bar{g} is “horizontally invariant”.)

For the Rayleigh quotient f , this yields

$$\begin{aligned} \overline{\nabla_{\eta} \text{grad } f} &= P_Y^h (D\overline{\text{grad } f}(Y) [\bar{\eta}_Y]) \\ &= 2 P_Y^h \left(A\bar{\eta}_Y - \bar{\eta}_Y (Y^T Y)^{-1} Y^T A Y \right). \end{aligned}$$

Newton's equation

Newton's equation at $\pi(Y)$ is

$$\nabla_{\eta_{\pi(Y)}} \text{grad } f = -\text{grad } f(\pi(Y))$$

for the unknown $\eta_{\pi(Y)} \in T_{\pi(Y)} \text{Grass}(p, n)$.

To turn this equation into a matrix equation, we take its horizontal lift.

This yields

$$P_Y^h \left(A\bar{\eta}_Y - \bar{\eta}_Y(Y^T Y)^{-1} Y^T A Y \right) = -P_Y^h A Y, \quad \bar{\eta}_Y \in \mathcal{H}_Y,$$

whose solution $\bar{\eta}_Y$ in the horizontal space \mathcal{H}_Y is the horizontal lift of the solution η of the Newton equation.

Retraction

Newton's method sends $\pi(Y)$ to \mathcal{Y}_+ according to

$$\begin{aligned}\nabla_{\eta_{\pi(Y)}} \text{grad } f &= -\text{grad } f(\pi(Y)) \\ \mathcal{Y}_+ &= R_{\pi(Y)}(\eta_{\pi(Y)}).\end{aligned}$$

It remains to pick the retraction R .

Choice: R defined by

$$R_{\pi(Y)}\xi_{\pi(Y)} = \pi(Y + \bar{\xi}_Y).$$

(This is a well-defined retraction.)

Newton's iteration for RQ on Grassmann

Require: Symmetric matrix A .

Input: Initial iterate $Y_0 \in \text{ST}(p, n)$.

Output: Sequence of iterates $\{Y_k\}$ in $\text{ST}(p, n)$.

- 1: **for** $k = 0, 1, 2, \dots$ **do**
- 2: Solve the linear system

$$\begin{cases} P_{Y_k}^h (AZ_k - Z_k(Y_k^T Y_k)^{-1} Y_k^T A Y_k) = -P_{Y_k}^h (A Y_k) \\ Y_k^T Z_k = 0 \end{cases}$$

for the unknown Z_k , where P_Y^h is the orthogonal projector onto \mathcal{H}_Y . (The condition $Y_k^T Z_k = 0$ expresses that Z_k belongs to the horizontal space \mathcal{H}_{Y_k} .)

- 3: Set

$$Y_{k+1} = (Y_k + Z_k)N_k$$

where N_k is a nonsingular $p \times p$ matrix chosen for normalization purposes.

- 4: **end for**

Trust-region methods on Riemannian manifolds

Motivating application: Mechanical vibrations

Mass matrix M , stiffness matrix K .

Equation of vibrations (for undamped discretized linear structures):

$$Kx = \omega^2 Mx$$

were

- ▶ ω is an angular frequency of vibration
- ▶ x is the corresponding mode of vibration

Task: find lowest modes of vibration.

Generalized eigenvalue problem

Given $n \times n$ matrices $A = A^T$ and $B = B^T \succ 0$, there exist v_1, \dots, v_n in \mathbb{R}^n and $\lambda_1 \leq \dots \leq \lambda_n$ in \mathbb{R} such that

$$Av_i = \lambda_i Bv_i$$

$$v_i^T Bv_j = \delta_{ij}.$$

Task: find $\lambda_1, \dots, \lambda_p$ and v_1, \dots, v_p .

We assume throughout that $\lambda_p < \lambda_{p+1}$.

Case $p = 1$: optimization in \mathbb{R}^n

$$Av_i = \lambda_i Bv_i$$

Consider the Rayleigh quotient

$$\tilde{f} : \mathbb{R}_*^n \rightarrow \mathbb{R} : f(y) = \frac{y^T Ay}{y^T By}$$

Invariance: $\tilde{f}(\alpha y) = \tilde{f}(y)$.

Stationary points of \tilde{f} : αv_i , for all $\alpha \neq 0$.

Minimizers of \tilde{f} : αv_1 , for all $\alpha \neq 0$.

Difficulty: the minimizers are not isolated.

Remedy: optimization on manifold.

Case $p = 1$: optimization on ellipsoid

$$\tilde{f} : \mathbb{R}_*^n \rightarrow \mathbb{R} : f(y) = \frac{y^T A y}{y^T B y}$$

Invariance: $\tilde{f}(\alpha y) = \tilde{f}(y)$.

Remedy 1:

- ▶ $\mathcal{M} := \{y \in \mathbb{R}^n : y^T B y = 1\}$, *submanifold* of \mathbb{R}^n .
- ▶ $f : \mathcal{M} \rightarrow \mathbb{R} : f(y) = y^T A y$.

Stationary points of f : $\pm v_1, \dots, \pm v_n$.

Minimizers of f : $\pm v_1$.

Case $p = 1$: optimization on projective space

$$\tilde{f} : \mathbb{R}_*^n \rightarrow \mathbb{R} : f(y) = \frac{y^T A y}{y^T B y}$$

Invariance: $\tilde{f}(\alpha y) = \tilde{f}(y)$.

Remedy 2:

- ▶ $[y] := y\mathbb{R} := \{y\alpha : \alpha \in \mathbb{R}\}$
- ▶ $\mathcal{M} := \mathbb{R}_*^n / \mathbb{R} = \{[y]\}$
- ▶ $f : \mathcal{M} \rightarrow \mathbb{R} : f([y]) := \tilde{f}(y)$

Stationary points of f : $[v_1], \dots, [v_n]$.

Minimizer of f : $[v_1]$.

Case $p \geq 1$: optimization on the Grassmann manifold

$$\tilde{f} : \mathbb{R}_*^{n \times p} \rightarrow \mathbb{R} : \tilde{f}(Y) = \text{trace} \left((Y^T B Y)^{-1} Y^T A Y \right)$$

Invariance: $\tilde{f}(YR) = \tilde{f}(Y)$.

Define:

- ▶ $[Y] := \{YR : R \in \mathbb{R}_*^{p \times p}\}, \quad Y \in \mathbb{R}_*^{n \times p}$
- ▶ $\mathcal{M} := \text{Grass}(p, n) := \{[Y]\}$
- ▶ $f : \mathcal{M} \rightarrow \mathbb{R} : f([Y]) := \tilde{f}(Y)$

Stationary points of f : $\text{span}\{v_{i_1}, \dots, v_{i_p}\}$.

Minimizer of f : $[Y] = \text{span}\{v_1, \dots, v_p\}$.

Optimization on Manifolds

- ▶ Luenberger [Lue73], Gabay [Gab82]: optimization on submanifolds of \mathbb{R}^n .
- ▶ Smith [Smi93, Smi94] and Udriște [Udr94]: optimization on general Riemannian manifolds (steepest descent, Newton, CG).
- ▶ ...
- ▶ PAA, Baker and Gallivan [ABG07]: trust-region methods on Riemannian manifolds.
- ▶ PAA, Mahony, Sepulchre [AMS08]: *Optimization Algorithms on Matrix Manifolds*, textbook.

The Problem : Leftmost Eigenpairs of Matrix Pencil

Given $n \times n$ matrix pencil (A, B) , $A = A^T$, $B = B^T \succ 0$ with (unknown) eigen-decomposition

$$A [v_1 | \dots | v_n] = B [v_1 | \dots | v_n] \text{diag}(\lambda_1, \dots, \lambda_n)$$

$$[v_1 | \dots | v_n]^T B [v_1 | \dots | v_n] = I, \quad \lambda_1 < \lambda_2 \leq \dots \leq \lambda_n.$$

The problem is to **compute the minor eigenvector $\pm v_1$** .

The ideal algorithm

Given (A, B) , $A = A^T$, $B = B^T \succ 0$ with (unknown) eigenvalues $0 < \lambda_1 \leq \dots \leq \lambda_n$ and associated eigenvectors v_1, \dots, v_n .

1. **Global convergence:**

- ▶ Convergence to some eigenvector for **all** initial conditions.
- ▶ **Stable** convergence to the “leftmost” eigenvector **$\pm v_1$ only**.

2. **Superlinear** (cubic) local convergence to $\pm v_1$.

The ideal algorithm

Given (A, B) , $A = A^T$, $B = B^T \succ 0$ with (unknown) eigenvalues $0 < \lambda_1 \leq \dots \leq \lambda_n$ and associated eigenvectors v_1, \dots, v_n .

1. **Global convergence:**
 - ▶ Convergence to some eigenvector for **all** initial conditions.
 - ▶ **Stable** convergence to the “leftmost” eigenvector $\pm v_1$ **only**.
2. **Superlinear** (cubic) local convergence to $\pm v_1$.
3. **“Matrix-free”** (no factorization of A, B)
but possible use of **preconditioner**.

The ideal algorithm

Given (A, B) , $A = A^T$, $B = B^T \succ 0$ with (unknown) eigenvalues $0 < \lambda_1 \leq \dots \leq \lambda_n$ and associated eigenvectors v_1, \dots, v_n .

1. **Global convergence:**
 - ▶ Convergence to some eigenvector for **all** initial conditions.
 - ▶ **Stable** convergence to the “leftmost” eigenvector $\pm v_1$ **only**.
2. **Superlinear** (cubic) local convergence to $\pm v_1$.
3. **“Matrix-free”** (no factorization of A, B)
but possible use of **preconditioner**.
4. **Minimal storage** space required.

Strategy

- ▶ Rewrite computation of leftmost eigenpair as an **optimization problem (on a manifold)**.

Strategy

- ▶ Rewrite computation of leftmost eigenpair as an **optimization problem (on a manifold)**.
- ▶ Use a **model-trust-region** scheme to solve the problem.
⇒ **Global convergence.**

Strategy

- ▶ Rewrite computation of leftmost eigenpair as an **optimization problem (on a manifold)**.
- ▶ Use a **model-trust-region** scheme to solve the problem.
 \rightsquigarrow **Global convergence.**
- ▶ Take the **exact quadratic model** (at least, close to the solution).
 \rightsquigarrow **Superlinear convergence.**

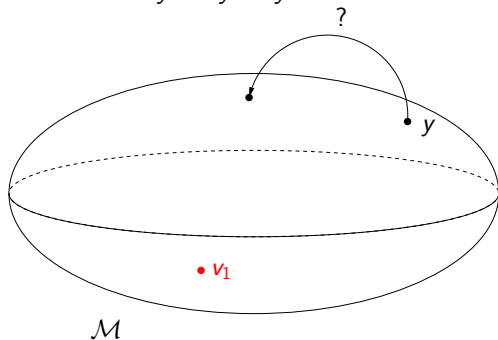
Strategy

- ▶ Rewrite computation of leftmost eigenpair as an **optimization problem (on a manifold)**.
- ▶ Use a **model-trust-region** scheme to solve the problem.
~> **Global convergence**.
- ▶ Take the **exact quadratic model** (at least, close to the solution).
~> **Superlinear convergence**.
- ▶ Solve the trust-region subproblems using the **(Steihaug-Toint) truncated CG (tCG)** algorithm.
~> **"Matrix-free"**, preconditioned iteration.
~> **Minimal storage** of iteration vectors.

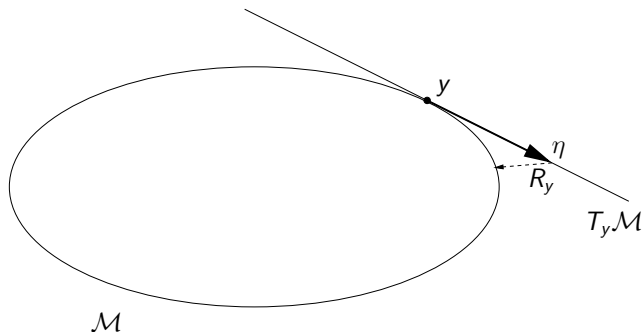
Iteration on the manifold

Manifold: ellipsoid $\mathcal{M} = \{y \in \mathbb{R}^n : y^T B y = 1\}$.

Cost function: $f : \mathcal{M} \rightarrow \mathbb{R} : y \mapsto y^T A y$



Tangent space and retraction (2D picture)



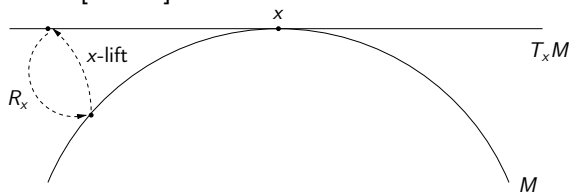
Tangent space: $T_y \mathcal{M} := \{\eta \in \mathbb{R}^n : y^T B \eta = 0\}$.

Retraction: $R_y \eta := (y + \eta) / \|y + \eta\|_B$.

Lifted cost function: $\hat{f}_y(\eta) := f(R_y \eta) = \frac{(y+\eta)^T A (y+\eta)}{(y+\eta)^T B (y+\eta)}$.

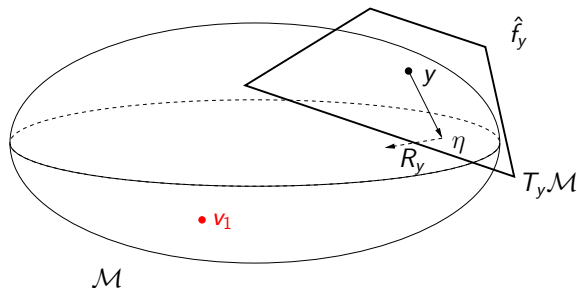
Concept of retraction

Introduced by Shub [Shu86].



1. R_x is defined and one-to-one in a neighbourhood of 0_x in $T_x M$.
2. $R_x(0_x) = x$.
3. $DR_x(0_x) = \text{id}_{T_x M}$, the identity mapping on $T_x M$, with the canonical identification $T_{0_x} T_x M \simeq T_x M$.

Tangent space and retraction



Tangent space: $T_y \mathcal{M} := \{\eta \in \mathbb{R}^n : y^T B \eta = 0\}$.

Retraction: $R_y \eta := (y + \eta) / \|y + \eta\|_B$.

Lifted cost function: $\hat{f}_y(\eta) := f(R_y \eta) = \frac{(y + \eta)^T A (y + \eta)}{(y + \eta)^T B (y + \eta)}$.

Quadratic model

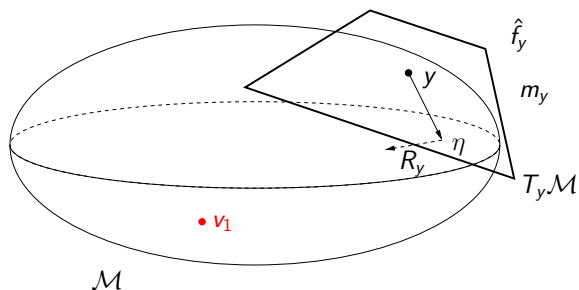
$$\begin{aligned}\hat{f}_y(\eta) &= \frac{y^T A y}{y^T B y} + 2 \frac{y^T A \eta}{y^T B y} + \frac{1}{y^T B y} \left(\eta^T A \eta - \frac{y^T A y}{y^T B y} \eta^T B \eta \right) + \dots \\ &= f(y) + 2 \langle P A y, \eta \rangle + \frac{1}{2} \langle 2P(A - f(y)B)P \eta, \eta \rangle + \dots\end{aligned}$$

where $\langle u, v \rangle = u^T v$ and $P = I - B y (y^T B^2 y)^{-1} y^T B$.

Model:

$$m_y(\eta) = f(y) + 2 \langle P A y, \eta \rangle + \frac{1}{2} \langle P(A - f(y)B)P \eta, \eta \rangle, \quad y^T B \eta = 0.$$

Quadratic model



$$m_y(\eta) = f(y) + 2\langle PAy, \eta \rangle + \frac{1}{2}\langle P(A - f(y)B)P\eta, \eta \rangle, \quad y^T B \eta = 0.$$

Newton vs Trust-Region

Model:

$$m_y(\eta) = f(y) + 2\langle PAy, \eta \rangle + \frac{1}{2}\langle P(A - f(y)B)P\eta, \eta \rangle, \quad y^T B\eta = 0. \quad (1)$$

Newton vs Trust-Region

Model:

$$m_y(\eta) = f(y) + 2\langle PAy, \eta \rangle + \frac{1}{2}\langle P(A - f(y)B)P\eta, \eta \rangle, \quad y^T B\eta = 0. \quad (1)$$

Newton method: Compute the **stationary point** of the model, i.e., solve

$$P(A - f(y)B)P\eta = -PAy.$$

Newton vs Trust-Region

Model:

$$m_y(\eta) = f(y) + 2\langle PAy, \eta \rangle + \frac{1}{2}\langle P(A - f(y)B)P\eta, \eta \rangle, \quad y^T B\eta = 0. \quad (1)$$

Newton method: Compute the **stationary point** of the model, i.e., solve

$$P(A - f(y)B)P\eta = -PAy.$$

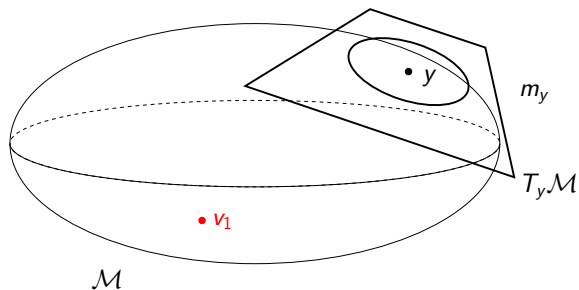
Instead, compute (approximately) the **minimizer** of m_y within a **trust-region**

$$\{\eta \in T_x\mathcal{M} : \eta^T \eta \leq \Delta^2\}.$$

Trust-region subproblem

Minimize

$$m_y(\eta) = f(y) + 2\langle PAy, \eta \rangle + \frac{1}{2}\langle P(A - f(y)B)P\eta, \eta \rangle, \quad y^T B \eta = 0.$$

subject to $\eta^T \eta \leq \Delta^2$.

Truncated CG method for the TR subproblem (1)

Let $\langle \cdot, \cdot \rangle$ denote the standard inner product and let $\mathcal{H}_{x_k} := P(A - f(x_k)B)P$ denote the Hessian operator.

Initializations:

Set $\eta_0 = 0$, $r_0 = P_{x_k} A x_k = A x_k - B x_k (x_k^T B^2 x_k)^{-1} x_k^T B A x_k$, $\delta_0 = -r_0$;

Then repeat the following loop on j :

Check for negative curvature

if $\langle \delta_j, \mathcal{H}_{x_k} \delta_j \rangle \leq 0$

 Compute τ such that $\eta = \eta_j + \tau \delta_j$ minimizes $m(\eta)$ in (1) and satisfies $\|\eta\| = \Delta$;

return η ;

Truncated CG method for the TR subproblem (2)

Generate next inner iterate

Set $\alpha_j = \langle r_j, r_j \rangle / \langle \delta_j, \mathcal{H}_{x_k} \delta_j \rangle$;

Set $\eta_{j+1} = \eta_j + \alpha_j \delta_j$;

Check trust-region

if $\|\eta_{j+1}\| \geq \Delta$

 Compute $\tau \geq 0$ such that $\eta = \eta_j + \tau \delta_j$ satisfies $\|\eta\| = \Delta$;

return η ;

Truncated CG method for the TR subproblem (3)

Update residual and search direction

Set $r_{j+1} = r_j + \alpha_j \mathcal{H}_{x_k} \delta_j$;

Set $\beta_{j+1} = \langle r_{j+1}, r_{j+1} \rangle / \langle r_j, r_j \rangle$;

Set $\delta_{j+1} = -r_{j+1} + \beta_{j+1} \delta_j$;

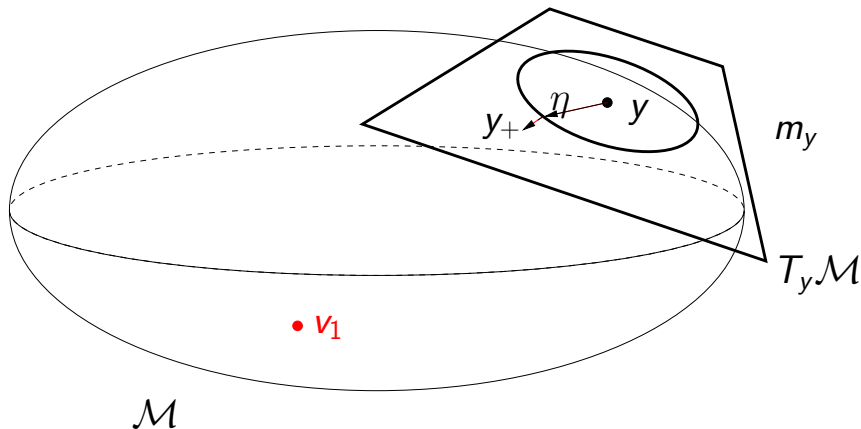
$j \leftarrow j + 1$;

Check residual

If $\|r_j\| \leq \|r_0\| \min(\|r_0\|^\theta, \kappa)$ for some prescribed θ and κ

return η_j ;

Overall iteration



The outer iteration – manifold trust-region (1)

Data: symmetric $n \times n$ matrices A and B , with B positive definite.

Parameters: $\bar{\Delta} > 0$, $\Delta_0 \in (0, \bar{\Delta})$, and $\rho' \in (0, \frac{1}{4})$.

Input: initial iterate $x_0 \in \{y : y^T B y = 1\}$.

Output: sequence of iterates $\{x_k\}$ in $\{y : y^T B y = 1\}$.

Initialization: $k = 0$

Repeat the following:

The outer iteration – manifold trust-region (2)

- ▶ Obtain η_k using the Steihaug-Toint truncated conjugate-gradient method to approximately solve the trust-region subproblem

$$\min_{x_k^T B \eta = 0} m_{x_k}(\eta) \quad \text{s.t.} \quad \|\eta\| \leq \Delta_k, \quad (2)$$

where m is defined in (1).

The outer iteration – manifold trust-region (3)

- Evaluate

$$\rho_k = \frac{\hat{f}_{x_k}(0) - \hat{f}_{x_k}(\eta_k)}{m_{x_k}(0) - m_{x_k}(\eta_k)} \quad (3)$$

where $\hat{f}_{x_k}(\eta) = \frac{(x_k + \eta)^T A(x_k + \eta)}{(x_k + \eta)^T B(x_k + \eta)}$.

- Update the trust-region radius:

if $\rho_k < \frac{1}{4}$

$$\Delta_{k+1} = \frac{1}{4} \Delta_k$$

else if $\rho_k > \frac{3}{4}$ **and** $\|\eta_k\| = \Delta_k$

$$\Delta_{k+1} = \min(2\Delta_k, \bar{\Delta})$$

else

$$\Delta_{k+1} = \Delta_k;$$

The outer iteration – manifold trust-region (4)

- Update the iterate:

if $\rho_k > \rho'$

$$x_{k+1} = (x_k + \eta_k) / \|x_k + \eta_k\|_B; \quad (4)$$

else

$$x_{k+1} = x_k;$$

$$k \leftarrow k + 1$$

Strategy

- ▶ Rewrite computation of leftmost eigenpair as an **optimization problem (on a manifold)**.
- ▶ Use a **model-trust-region** scheme to solve the problem.
~> **Global convergence**.
- ▶ Take the **exact quadratic model** (at least, close to the solution).
~> **Superlinear convergence**.
- ▶ Solve the trust-region subproblems using the **(Steihaug-Toint) truncated CG (tCG)** algorithm.
~> **"Matrix-free"**, preconditioned iteration.
~> **Minimal storage** of iteration vectors.

Summary

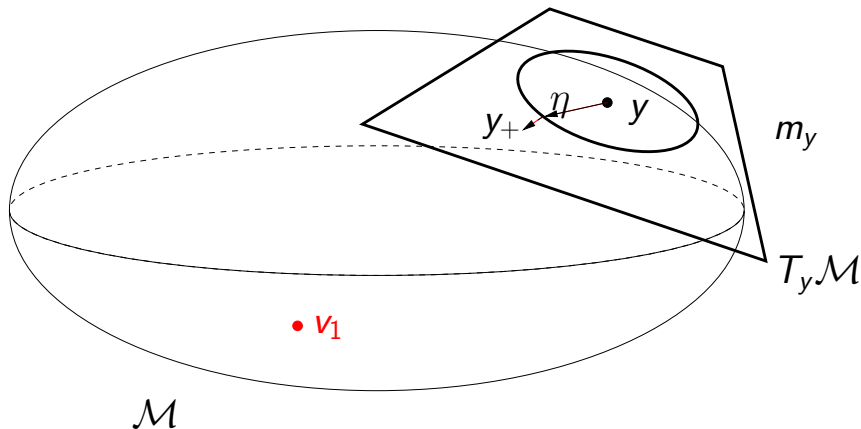
We have obtained a trust-region algorithm for minimizing the Rayleigh quotient over an ellipsoid.

Summary

We have obtained a trust-region algorithm for minimizing the Rayleigh quotient over an ellipsoid.

Generalization to trust-region algorithms for minimizing functions on manifolds: the **Riemannian Trust-Region (RTR)** method [ABG07].

Convergence analysis



Global convergence of Riemannian Trust-Region algorithms

Let $\{x_k\}$ be a sequence of iterates generated by the RTR algorithm with $\rho' \in (0, \frac{1}{4})$. Suppose that f is C^2 and bounded below on the level set $\{x \in M : f(x) < f(x_0)\}$. Suppose that $\|\text{grad } f(x)\| \leq \beta_g$ and $\|\text{Hess } f(x)\| \leq \beta_H$ for some constants β_g, β_H , and all $x \in M$. Moreover suppose that

$$\left\| \frac{D}{dt} \frac{d}{dt} R_t \xi \right\| \leq \beta_D \quad (5)$$

for some constant β_D , for all $\xi \in TM$ with $\|\xi\| = 1$ and all $t < \delta_D$, where $\frac{D}{dt}$ denotes the covariant derivative along the curve $t \mapsto R_t \xi$. Further suppose that all approximate solutions η_k of the trust-region subproblems produce a decrease of the model that is at least a fixed fraction of the Cauchy decrease.

Global convergence (cont'd)

It then follows that

$$\lim_{k \rightarrow \infty} \text{grad } f(x_k) = 0.$$

And only the local minima are stable (the saddle points and local maxima are unstable).

Local convergence of Riemannian Trust-Region algorithms

Consider the RTR-tCG algorithm. Suppose that f is a C^2 cost function on M and that

$$\|\mathcal{H}_k - \text{Hess } \hat{f}_{x_k}(0_k)\| \leq \beta_{\mathcal{H}} \|\text{grad } f(x_k)\|. \quad (6)$$

Let $v \in M$ be a **nondegenerate local minimum** of f , (i.e., $\text{grad } f(v) = 0$ and $\text{Hess } f(v)$ is positive definite). Further assume that $\text{Hess } \hat{f}_{x_k}$ is Lipschitz-continuous at 0_x uniformly in x in a neighborhood of v , i.e., there exist $\beta_1 > 0$, $\delta_1 > 0$ and $\delta_2 > 0$ such that, for all $x \in B_{\delta_1}(v)$ and all $\xi \in B_{\delta_2}(0_x)$, it holds

$$\|\text{Hess } \hat{f}_{x_k}(\xi) - \text{Hess } \hat{f}_{x_k}(0_{x_k})\| \leq \beta_{L2} \|\xi\|. \quad (7)$$

Local convergence (cont'd)

Then there exists $c > 0$ such that, for all sequences $\{x_k\}$ generated by the RTR-tCG algorithm converging to v , there exists $K > 0$ such that for all $k > K$,

$$\text{dist}(x_{k+1}, v) \leq c (\text{dist}(x_k, v))^{\min\{\theta+1, 2\}}, \quad (8)$$

where θ governs the stopping criterion of the tCG inner iteration.

Convergence of trust-region-based eigensolver

Theorem:

Let (A, B) be an $n \times n$ symmetric/positive-definite matrix pencil with eigenvalues $\lambda_1 < \lambda_2 \leq \dots \leq \lambda_{n-1} \leq \lambda_n$ and an associated B -orthonormal basis of eigenvectors (v_1, \dots, v_n) .

Let $\mathcal{S}_i = \{y : Ay = \lambda_i By, y^T By = 1\}$ denote the intersection of the eigenspace of (A, B) associated to λ_i with the set $\{y : y^T By = 1\}$.

...

Convergence (global)

- (i) Let $\{x_k\}$ be a sequence of iterates generated by the Algorithm. Then $\{x_k\}$ converges to the eigenspace of (A, B) associated to one of its eigenvalues. That is, there exists i such that $\lim_{k \rightarrow \infty} \text{dist}(x_k, \mathcal{S}_i) = 0$.
- (ii) Only the set $\mathcal{S}_1 = \{\pm v_1\}$ is stable.

Convergence (local)

(iii) There exists $c > 0$ such that, for all sequences $\{x_k\}$ generated by the Algorithm converging to \mathcal{S}_1 , there exists $K > 0$ such that for all $k > K$,

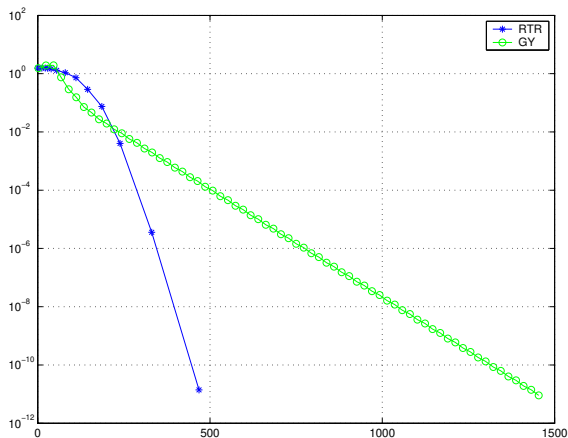
$$\text{dist}(x_{k+1}, \mathcal{S}_1) \leq c (\text{dist}(x_k, \mathcal{S}_1))^{\min\{\theta+1, 2\}} \quad (9)$$

with $\theta > 0$.

Strategy

- ▶ Rewrite computation of leftmost eigenpair as an **optimization problem (on a manifold)**.
- ▶ Use a **model-trust-region** scheme to solve the problem.
~> **Global convergence**.
- ▶ Take the **exact quadratic model** (at least, close to the solution).
~> **Superlinear convergence**.
- ▶ Solve the trust-region subproblems using the **(Steihaug-Toint) truncated CG (tCG)** algorithm.
~> **"Matrix-free"**, preconditioned iteration.
~> **Minimal storage** of iteration vectors.

Numerical experiments: RTR vs Krylov [GY02]



Distance to target versus matrix-vector multiplications.
Symmetric/positive-definite generalized eigenvalue problem.

A new tool for Optimization On
Manifolds:
Vector Transport

Filling a gap

	Purely Riemannian way	Pragmatic way
Update	Search along the geodesic tangent to the search direction	Search along any curve tangent to the search direction (prescribed by a <i>retraction</i>)
Displacement of tgt vectors	Parallel translation induced by $\overset{g}{\nabla}$??

Where do we use parallel translation?

In [CG](#). Quoting (approximately) Smith (1994):

1. Select $x_0 \in \mathcal{M}$, compute $\eta_0 = -\text{grad } f(x_0)$, and set $k = 0$
2. Compute t_k such that $f(\text{Exp}_{x_k}(t_k \eta_k)) \leq f(\text{Exp}_{x_k}(t \eta_k))$ for all $t \geq 0$.
3. Set $x_{k+1} = \text{Exp}_{x_k}(t_k \eta_k)$.
4. Set $\eta_{k+1} = -\text{grad } f(x_{k+1}) + \beta_{k+1} \tau \eta_k$, where τ is the **parallel translation** along the geodesic from x_k to x_{k+1} . Increment k and go to step 2.

Where do we use parallel translation?

In **BFGS**. Quoting (approximately) Gabay (1982):

$x_{k+1} = \text{Exp}_{x_k}(t_k \xi_k)$ (update along geodesic)

$\text{grad } f(x_{k+1}) - \tau_0^{t_k} \text{grad } f(x_k) = B_{k+1} \tau_0^{t_k}(t_k \xi_k)$ (requirement on approximate Jacobian B)

This leads to the a *generalized BFGS update formula* involving parallel translation.

Where else could we use parallel translation?

In **finite-difference quasi-Newton**.

Let ξ be a vector field on a Riemannian manifold \mathcal{M} . Exact Jacobian of ξ at $x \in \mathcal{M}$: $J_\xi(x)[\eta] = \nabla_\eta \xi$.

Finite difference approximation to J_ξ : choose a basis (E_1, \dots, E_d) of $T_x \mathcal{M}$ and define $\tilde{J}(x)$ as the linear operator that satisfies

$$\tilde{J}(x)[E_i] = \frac{\tau_h^0 \xi_{\text{Exp}_x(hE_i)} - \xi_x}{h}.$$

Filling a gap

	Purely Riemannian way	Pragmatic way
Update	Search along the geodesic tangent to the search direction	Search along any prescribed curve tangent to the search direction
Displacement of tgt vectors	Parallel translation induced by $\overset{g}{\nabla}$??

Parallel translation can be tough

Edelman et al (1998): We are unaware of any closed form expression for the parallel translation on the Stiefel manifold (defined with respect to the Riemannian connection induced by the embedding in $\mathbb{R}^{n \times p}$).

Parallel transport along geodesics on Grassmannians:

$$\overline{\xi(t)}_{Y(t)} = -Y_0 V \sin(\Sigma t) U^T \overline{\xi(0)}_{Y_0} + U \cos(\Sigma t) U^T \overline{\xi(0)}_{Y_0} + (I - UU^T) \overline{\xi(0)}_{Y_0}.$$

where $\overline{\dot{Y}(0)}_{Y_0} = U \Sigma V^T$ is a thin SVD.

Alternatives found in the literature

Edelman et al (1998): “extrinsic” CG algorithm. “Tangency of the search direction at the new point is imposed via the projection $I - YY^T$ ” (instead of via parallel translation).

Brace & Manton (2006), *An improved BFGS-on-manifold algorithm for computing weighted low rank approximation*. “The second change is that parallel translation is not defined with respect to the Levi-Civita connection, but rather is all but ignored.”

Filling a gap

	Purely Riemannian way	Pragmatic way
Update	Search along the geodesic tangent to the search direction	Search along any curve tangent to the search direction (prescribed by a <i>retraction</i>)
Displacement of tgt vectors	Parallel translation induced by $\overset{g}{\nabla}$??

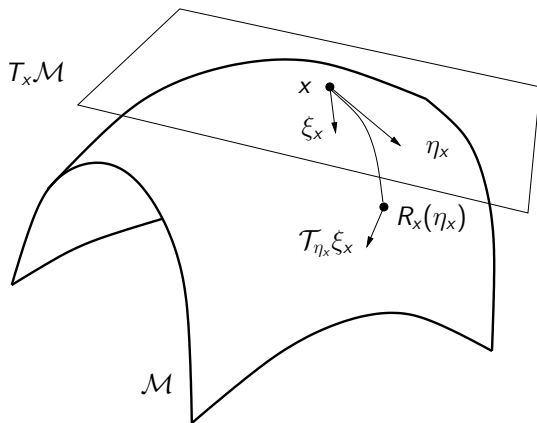
Filling a gap: Vector Transport

	Purely Riemannian way	Pragmatic way
Update	Search along the geodesic tangent to the search direction	Search along any curve tangent to the search direction (prescribed by a <i>retraction</i>)
Displacement of tgt vectors	Parallel translation induced by $\overset{g}{\nabla}$	Vector Transport

Still to come

- ▶ Vector transport in one picture
- ▶ Formal definition
- ▶ Particular vector transports
- ▶ Applications: finite-difference Newton, BFGS, CG.

The concept of vector transport



Retraction

A *retraction* on a manifold \mathcal{M} is a smooth mapping

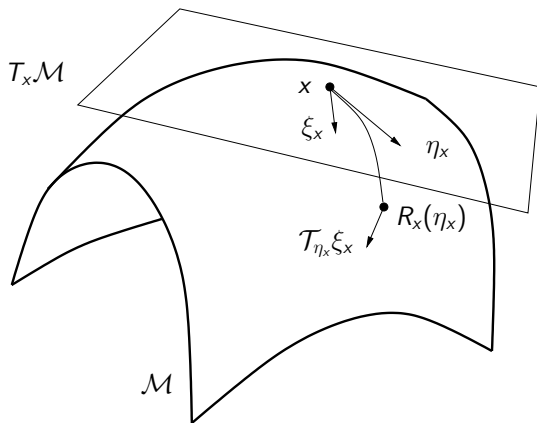
$$R : T\mathcal{M} \rightarrow \mathcal{M}$$

such that

1. $R(0_x) = x$ for all $x \in \mathcal{M}$, where 0_x denotes the origin of $T_x\mathcal{M}$;
2. $\frac{d}{dt}R(t\xi_x)|_{t=0} = \xi_x$ for all $\xi_x \in T_x\mathcal{M}$.

Consequently, the curve $t \mapsto R(t\xi_x)$ is a curve on \mathcal{M} tangent to ξ_x .

The concept of vector transport – Whitney sum



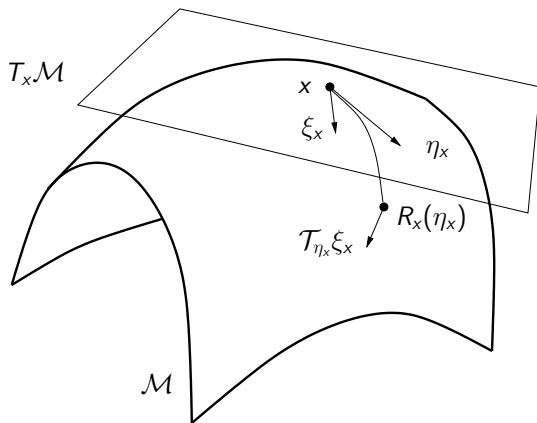
Whitney sum

Let $T\mathcal{M} \oplus T\mathcal{M}$ denote the set

$$T\mathcal{M} \oplus T\mathcal{M} = \{(\eta_x, \xi_x) : \eta_x, \xi_x \in T_x\mathcal{M}, x \in \mathcal{M}\}.$$

This set admits a natural manifold structure.

The concept of vector transport – definition



Vector transport: definition

A *vector transport* on a manifold \mathcal{M} on top of a retraction R is a smooth map

$$T\mathcal{M} \oplus T\mathcal{M} \rightarrow T\mathcal{M} : (\eta_x, \xi_x) \mapsto \mathcal{T}_{\eta_x}(\xi_x) \in T\mathcal{M}$$

satisfying the following properties for all $x \in \mathcal{M}$:

1. (Underlying retraction) $\mathcal{T}_{\eta_x}\xi_x$ belongs to $T_{R_x(\eta_x)}\mathcal{M}$.
2. (Consistency) $\mathcal{T}_{0_x}\xi_x = \xi_x$ for all $\xi_x \in T_x\mathcal{M}$;
3. (Linearity) $\mathcal{T}_{\eta_x}(a\xi_x + b\zeta_x) = a\mathcal{T}_{\eta_x}(\xi_x) + b\mathcal{T}_{\eta_x}(\zeta_x)$.

Inverse vector transport

When it exists, $(\mathcal{T}_{\eta_x})^{-1}(\xi_{R_x(\eta_x)})$ belongs to $T_x\mathcal{M}$. If η and ξ are two vector fields on \mathcal{M} , then $(\mathcal{T}_\eta)^{-1}\xi$ is naturally defined as the vector field satisfying

$$((\mathcal{T}_\eta)^{-1}\xi)_x = (\mathcal{T}_{\eta_x})^{-1}(\xi_{R_x(\eta_x)}).$$

Still to come

- ▶ Vector transport in one picture
- ▶ Formal definition
- ▶ Particular vector transports
- ▶ Applications: finite-difference Newton, BFGS, CG.

Parallel translation is a vector transport

Proposition

If ∇ is an affine connection and R is a retraction on a manifold \mathcal{M} , then

$$\mathcal{T}_{\eta_x}(\xi_x) := P_{\gamma}^{1 \leftarrow 0} \xi_x \quad (10)$$

is a vector transport with associated retraction R , where P_{γ} denotes the parallel translation induced by ∇ along the curve $t \mapsto \gamma(t) = R_x(t\eta_x)$.

Vector transport on Riemannian submanifolds

If \mathcal{M} is an embedded submanifold of a Euclidean space \mathcal{E} and \mathcal{M} is endowed with a retraction R , then we can rely on the natural inclusion $T_y\mathcal{M} \subset \mathcal{E}$ for all $y \in \mathcal{N}$ to simply define the vector transport by

$$\mathcal{T}_{\eta_x}\xi_x := P_{R_x(\eta_x)}\xi_x, \quad (11)$$

where P_x denotes the orthogonal projector onto $T_x\mathcal{N}$.

Still to come

- ▶ Vector transport in one picture
- ▶ Formal definition
- ▶ Particular vector transports
- ▶ Applications: finite-difference Newton, BFGS, CG.

Vector transport in finite differences

Let \mathcal{M} be a manifold endowed with a vector transport \mathcal{T} on top of a retraction R . Let $x \in \mathcal{M}$ and let (E_1, \dots, E_d) be a basis of $T_x\mathcal{M}$. Given a smooth vector field ξ and a real constant $h > 0$, let $\tilde{J}_\xi(x) : T_x\mathcal{M} \rightarrow T_x\mathcal{M}$ be the linear operator that satisfies, for $i = 1, \dots, d$,

$$\tilde{J}_\xi(x)[E_i] = \frac{(\mathcal{T}_{hE_i})^{-1}\xi_{R(hE_i)} - \xi_x}{h}. \quad (12)$$

Lemma (finite differences)

Let x_* be a nondegenerate zero of ξ . Then there is $c > 0$ such that, for all x sufficiently close to x_* and all h sufficiently small, it holds that

$$\|\tilde{J}_\xi(x)[E_i] - J(x)[E_i]\| \leq c(h + \|\xi_x\|). \quad (13)$$

Convergence of Newton's method with finite differences

Proposition

Consider the geometric Newton method where the exact Jacobian $J(x_k)$ is replaced by the operator $\tilde{J}_\xi(x_k)$ with $h := h_k$. If

$$\lim_{k \rightarrow \infty} h_k = 0,$$

then the convergence to nondegenerate zeros of ξ is superlinear. If, moreover, there exists some constant c such that

$$h_k \leq c \|\xi_{x_k}\|$$

for all k , then the convergence is (at least) quadratic.

Vector transport in BFGS

With the notation

$$s_k := \mathcal{T}_{\eta_k} \eta_k \in T_{x_{k+1}} \mathcal{M},$$

$$y_k := \text{grad } f(x_{k+1}) - \mathcal{T}_{\eta_k}(\text{grad } f(x_k)) \in T_{x_{k+1}} \mathcal{M},$$

we define the operator $A_{k+1} : T_{x_{k+1}} \mathcal{M} \mapsto T_{x_{k+1}} \mathcal{M}$ by

$$A_{k+1} \eta = \tilde{A}_k \eta - \frac{\langle s_k, \tilde{A}_k \eta \rangle}{\langle s_k, \tilde{A}_k s_k \rangle} \tilde{A}_k s_k + \frac{\langle y_k, \eta \rangle}{\langle y_k, s_k \rangle} y_k \quad \text{for all } \eta \in T_{x_{k+1}} \mathcal{M},$$

with

$$\tilde{A}_k = \mathcal{T}_{\eta_k} \circ A_k \circ (\mathcal{T}_{\eta_k})^{-1}.$$

Vector transport in CG

Compute a step size α_k and set

$$x_{k+1} = R_{x_k}(\alpha_k \eta_k). \quad (14)$$

Compute β_{k+1} and set

$$\eta_{k+1} = -\text{grad } f(x_{k+1}) + \beta_{k+1} \mathcal{T}_{\alpha_k \eta_k}(\eta_k). \quad (15)$$

Filling a gap: Vector Transport

	Purely Riemannian way	Pragmatic way
Update	Search along the geodesic tangent to the search direction	Search along any curve tangent to the search direction (prescribed by a <i>retraction</i>)
Displacement of tgt vectors	Parallel translation induced by $\overset{g}{\nabla}$	Vector Transport

Ongoing work

- ▶ Use vector transport wherever we can.
- ▶ Extend convergence analyses.
- ▶ Develop recipes for building efficient vector transports.

BFGS Algorithm on Manifolds

- ▶ Main source: *Riemannian BFGS algorithm with applications*. Chunhong Qi, Kyle A. Gallivan, P.-A. Absil. Recent Advances in Optimization and its Applications in Engineering, Springer-Verlag, pp. 183-192, 2010. URL: http://www.inma.ucl.ac.be/~absil/Publi/Qi_RBFGS.htm
- ▶ Recent point of entry to the literature: *A Riemannian BFGS Method without Differentiated Retraction for Nonconvex Optimization Problems*. Wen Huang, P.-A. Absil, K. A. Gallivan. SIAM Journal on Optimization, 28:1, pp. 470-495, 2018. URL: <https://www.math.fsu.edu/~whuang2/papers/ARBMDRNOP.htm>

A (questionable) historical overview

	In \mathbb{R}^n	On Riemannian manifolds	
		using classical objects	using novel objects
Steepest descent	1966 (Armijo backtracking)	1972 (Luenberger)	1986–2008 ?
Newton	1740 (Simpson)	1993 (Smith)	2002 (Adler et al.)
Conjugate Grad	1964 (Fletcher–Reeves)	1993 (Smith)	2008 (PAA, Mahony, Sepulchre) ?
Trust regions	1985 (name created by Celis, Dennis, Tapia)	2007 (PAA, Baker, Gallivan)	2007 (PAA, Baker, Gallivan)
BFGS	1970 (B-F-G-S)	1982 (Gabay)	2010 (Qi, Gallivan, PAA)

Background on classical BFGS

- ▶ BFGS stands for Broyden–Fletcher–Goldfarb–Shanno.
- ▶ BFGS is a *quasi-Newton method*, where the Hessian found in the pure Newton is replaced by an approximation \mathcal{B}_k .
- ▶ The approximation \mathcal{B}_k undergoes a rank-two update at each iteration and satisfies the *secant condition*:

$$\mathcal{B}_{k+1}(x_{k+1} - x_k) = \text{grad } f(x_{k+1}) - \text{grad } f(x_k).$$

Symmetric secant update (PSB)

- ▶ Let $s_k = x_{k+1} - x_k$ and $y_k = \text{grad } f(x_{k+1}) - \text{grad } f(x_k)$. Then the secant condition becomes

$$\mathcal{B}_{k+1}s_k = y_k.$$

- ▶ What is \mathcal{B}_{k+1} that minimizes $\|\mathcal{B}_{k+1} - \mathcal{B}_k\|_F$ subject to $\mathcal{B}_{k+1}s_k = y_k$ and $\mathcal{B}_{k+1} - \mathcal{B}_k$ symmetric?

Answer given by the *symmetric secant update*, also called *Powell-symmetric-Broyden* (PSB) update:

$$\mathcal{B}_{k+1} = \mathcal{B}_k + \frac{(y_k - \mathcal{B}_k s_k) s_k^T + s_k^T (y_k - \mathcal{B}_k s_k)^T}{s_k^T s_k} - \frac{\langle y_k - \mathcal{B}_k s_k, s_k \rangle s_k s_k^T}{(s_k^T s_k)^2}$$

- ▶ Drawback: \mathcal{B}_{k+1} is not necessarily positive-definite. Hence the next search direction $\eta_k = -\mathcal{B}_k^{-1} \text{grad } f(x_k)$ may not be a descent direction.

Positive-definite secant update (BFGS)

- ▶ Let $s_k = x_{k+1} - x_k$ and $y_k = \text{grad } f(x_{k+1}) - \text{grad } f(x_k)$. Then the secant condition becomes

$$\mathcal{B}_{k+1}s_k = y_k.$$

- ▶ Let also $\mathcal{B}_k = LL^T$ be the Cholesky factorization.
- ▶ What is $\mathcal{B}_{k+1} = JJ^T$ with J nonsingular (guaranties \mathcal{B}_{k+1} symmetric positive definite) such that $\mathcal{B}_{k+1}s_k = y_k$ and $\|J - L\|_F$ as small as possible?

Answer given by the *positive definite secant update*, discovered independently by Broyden, Fletcher, Goldfarb and Shanno (BFGS) in 1970:

$$\mathcal{B}_{k+1} = \mathcal{B}_k + \frac{y_k y_k^T}{y_k^T s_k} - \frac{\mathcal{B}_k s_k (\mathcal{B}_k s_k)^T}{s_k^T \mathcal{B}_k s_k},$$

iff $s_k^T y_k > 0$. Otherwise, no solution.

Formulation of classical BFGS (in \mathbb{R}^n)

Algorithm 1 The classical BFGS algorithm (in \mathbb{R}^n)

- 1: Given: real-valued function f on \mathbb{R}^n ; initial iterate $\mathbf{x}_1 \in \mathbb{R}^n$; initial Hessian approximation \mathcal{B}_1 ;
 - 2: **for** $k = 1, 2, \dots$ **do**
 - 3: Obtain $\eta_k \in \mathbb{R}^n$ by solving: $\eta_k = -\mathcal{B}_k^{-1} \text{grad } f(\mathbf{x}_k)$.
 - 4: Perform a line search to obtain a step size α_k and set $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \eta_k$.
 - 5: Set $s_k := \alpha_k \eta_k$
 - 6: Set $y_k := \text{grad } f(\mathbf{x}_{k+1}) - \text{grad } f(\mathbf{x}_k)$
 - 7:
$$\mathcal{B}_{k+1} = \mathcal{B}_k + \frac{y_k y_k^T}{y_k^T s_k} - \frac{\mathcal{B}_k s_k (\mathcal{B}_k s_k)^T}{s_k^T \mathcal{B}_k s_k}.$$
 - 8: **end for**
-

Significant Riemannian Manifolds

Sphere S^{n-1}

The manifold of unit sphere:

$$S^{n-1} = \{x \in \mathbb{R}^n : x^T x = 1\}$$

Compact Stiefel Manifold

The manifold of orthonormal bases:

$$\text{St}((, p), n) = \{Q \in \mathbb{R}^{n \times p} : Q^T Q = I_p\}$$

Grassmann manifold

Manifold of linear subspaces:

$$\text{Grass}((, k), n) = \{k\text{-dimensional subspaces of } \mathbb{R}^n\}$$

Applications

- ▶ computing the leftmost eigenvector of A (S^{n-1})

$$f : S^{n-1} \rightarrow \mathbb{R} : x \mapsto x^T A x, A = A^T$$

- ▶ Procrustes Problem ($\text{St}(p, n)$)

$$f : \text{St}(p, n) \rightarrow \mathbb{R} : Q \mapsto \|AQ - QB\|_F, A : n \times n, B : p \times p$$

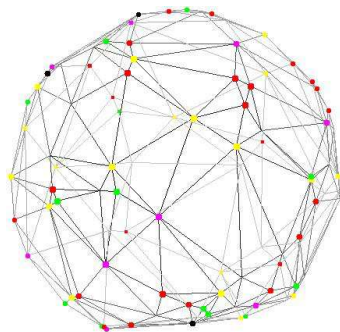
Application

- ▶ Thomson Problem
($S^{n-1} \times \dots \times S^{n-1}$)

- ▶ $f : [x_1, x_2, \dots, x_N] \mapsto$

$$\sum_{\substack{i,j=1 \\ i \neq j}}^N \frac{1}{\|x_i - x_j\|^2}$$

- ▶ Optimally arrange N repulsive particles on a sphere
- ▶ Determining the minimum energy configuration of these particles



Applet: <http://thomson.phy.syr.edu/thomsonapplet.htm>

- The weighted low rank approximation problem on $\text{Grass}(n, k)$:

$$\min_{\substack{R \in \mathbb{R}^{p \times n} \\ \text{rank}\{R\} \leq r}} \|X - R\|_Q^2 \quad (16)$$

$X \in \mathbb{R}^{p \times n}$: a given data matrix, $Q \in \mathbb{R}^{pn \times pn}$: a weighted matrix,
 $\|X - R\|_Q^2 = \text{vec}\{X - R\}^T Q \text{vec}\{X - R\}$., rewrite (16) as

$$\min_{\substack{N \in \mathbb{R}^{n \times (n-r)} \\ N^T N = I}} \min_{\substack{R \in \mathbb{R}^{p \times n} \\ RN = 0}} \|X - R\|_Q^2$$

The inner minimization has a closed form solution, call it $f(N)$:

$$f(N) = \text{vec}\{X\}^T (N \otimes I_p) \left[(N \otimes I_p)^T Q^{-1} (N \otimes I_p) \right]^{-1} (N \otimes I_p)^T \text{vec}\{X\}$$

Riemannian BFGS: past and future

Previous work on BFGS on manifolds

- ▶ Gabay [Gab82] discussed a version using parallel translation
- ▶ Brace and Manton restrict themselves to a version on the Grassmann manifold and the problem of weighted low-rank approximations [BM06].
- ▶ Savas and Lim apply a version to the more complicated problem of best multilinear approximations with tensors on a product of Grassmann manifolds [SL10].

Our goals

- ▶ Make the algorithm faster.
- ▶ Understand its convergence better.

Riemannian BFGS: a glimpse of the algorithm

- 1: Given: **Riemannian manifold** (M, g) ; **vector transport** \mathcal{T} on M with associated **retraction** R ; real-valued function f on M ; initial iterate $\mathbf{x}_1 \in M$; initial Hessian approximation \mathcal{B}_1 ;
- 2: **for** $k = 1, 2, \dots$ **do**
- 3: Obtain $\eta_k \in T_{\mathbf{x}_k} M$ by solving: $\eta_k = -\mathcal{B}_k^{-1} \text{grad } f(\mathbf{x}_k)$.
- 4: Perform a line search on $\mathbb{R} \ni \alpha \mapsto f(R_{\mathbf{x}_k}(\alpha\eta_k)) \in \mathbb{R}$ to obtain a step size α_k ; set $\mathbf{x}_{k+1} = R_{\mathbf{x}_k}(\alpha_k\eta_k)$.
- 5: Define $s_k = \mathcal{T}_{\alpha_k\eta_k} \alpha_k\eta_k$ and $y_k = \text{grad } f(\mathbf{x}_{k+1}) - \mathcal{T}_{\alpha_k\eta_k} \text{grad } f(\mathbf{x}_k)$
- 6: Define the linear operator $\mathcal{B}_{k+1} : T_{\mathbf{x}_{k+1}} M \rightarrow T_{\mathbf{x}_{k+1}} M$ as follows

$$\mathcal{B}_{k+1}p = \tilde{\mathcal{B}}_k p - \frac{g(s_k, \tilde{\mathcal{B}}_k p)}{g(s_k, \tilde{\mathcal{B}}_k s_k)} \tilde{\mathcal{B}}_k s_k + \frac{g(y_k, p)}{g(y_k, s_k)} y_k, \quad \forall p \in T_{\mathbf{x}_{k+1}} M$$

$$\text{with } \tilde{\mathcal{B}}_k = \mathcal{T}_{\alpha_k\eta_k} \circ \mathcal{B}_k \circ (\mathcal{T}_{\alpha_k\eta_k})^{-1}$$

- 7: **end for**

Vector transport

Manifold algorithms

- ▶ Conjugate gradients
- ▶ Secant methods
- ▶ BFGS

where parallel translation is used to combine two or more tangent vectors from distinct tangent spaces.

Vector transport

We define a **vector transport** on a manifold \mathcal{M} to be a smooth mapping

$$T\mathcal{M} \oplus T\mathcal{M} \rightarrow T\mathcal{M} : (\eta_x, \xi_x) \mapsto \mathcal{T}_{\eta_x}(\xi_x) \in T\mathcal{M}$$

satisfying three properties for all $x \in \mathcal{M}$.

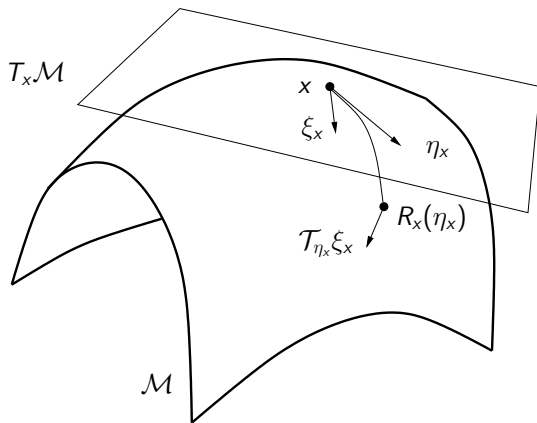


Figure: Vector transport.

Vector Transport

- ▶ (Associated retraction) There exists a retraction R , called the *retraction associated with \mathcal{T}* , such that the following diagram commutes

$$\begin{array}{ccc}
 (\eta_x, \xi_x) & \xrightarrow{\mathcal{T}} & \mathcal{T}_{\eta_x}(\xi_x) \\
 \downarrow & & \downarrow \pi \\
 \eta_x & \xrightarrow{R} & \pi(\mathcal{T}_{\eta_x}(\xi_x))
 \end{array}$$

where $\pi(\mathcal{T}_{\eta_x}(\xi_x))$ denotes the foot of the tangent vector $\mathcal{T}_{\eta_x}(\xi_x)$.

- ▶ (Consistency) $\mathcal{T}_{0_x}\xi_x = \xi_x$ for all $\xi_x \in T_x\mathcal{M}$;
- ▶ (Linearity) $\mathcal{T}_{\eta_x}(a\xi_x + b\zeta_x) = a\mathcal{T}_{\eta_x}(\xi_x) + b\mathcal{T}_{\eta_x}(\zeta_x)$.

Vector transport by differentiated retraction

Let M be a manifold endowed with retraction R , a particular vector transport is given by

$$\begin{aligned}\mathcal{T}_{\eta_x} \xi_x &:= DR_x(\eta_x)[\xi_x]; \text{ i.e.,} \\ \mathcal{T}_{\eta_x} \xi_x &:= \left. \frac{d}{dt} R_x(\eta_x + t\xi_x) \right|_{t=0};\end{aligned}$$

Vector transport by projection [AMS08, §8.1.2] (submanifolds only)

If M is an **embedded submanifold** of a Euclidean space ε and M is endowed with a retraction R , then

$$\mathcal{T}_{\eta_x} \xi_x := P_{R_x(\eta_x)} \xi_x,$$

where P_x denotes the orthogonal projector onto $T_x M$, is a vector transport.

Vector transport on quotient manifold

$\mathcal{M} = \overline{\mathcal{M}} / \sim$: a quotient manifold, where \mathcal{M} is an open subset of a Euclidean space ε .

$$\overline{(\mathcal{T}_{\eta_x} \xi_x)}_{\bar{x} + \bar{\eta}_{\bar{x}}} := P_{\bar{x} + \bar{\eta}_{\bar{x}}}^h \bar{\xi}_{\bar{x}},$$

where $P_{\bar{x}}^h Z : T_{\bar{x}} \overline{\mathcal{M}} \rightarrow \mathcal{H}_{\bar{x}}$ denotes the projection parallel to the vertical space $\mathcal{V}_{\bar{x}}$ onto the horizontal space $\mathcal{H}_{\bar{x}}$ at \bar{x} .

Algorithm 2 The Riemannian BFGS (RBFGS) algorithm

- 1: Given: **Riemannian manifold** (M, g) ; **vector transport** \mathcal{T} on M with associated **retraction** R ; real-valued function f on M ; initial iterate $\mathbf{x}_1 \in M$; initial Hessian approximation \mathcal{B}_1 ;
- 2: **for** $k = 1, 2, \dots$ **do**
- 3: Obtain $\eta_k \in T_{\mathbf{x}_k} M$ by solving: $\eta_k = -\mathcal{B}_k^{-1} \text{grad } f(\mathbf{x}_k)$.
- 4: Perform a line search on $\mathbb{R} \ni \alpha \mapsto f(R_{\mathbf{x}_k}(\alpha\eta_k)) \in \mathbb{R}$ to obtain a step size α_k ; set $\mathbf{x}_{k+1} = R_{\mathbf{x}_k}(\alpha_k\eta_k)$.
- 5: Define $s_k = \mathcal{T}_{\alpha_k\eta_k} \alpha_k\eta_k$ and $y_k = \text{grad } f(\mathbf{x}_{k+1}) - \mathcal{T}_{\alpha_k\eta_k} \text{grad } f(\mathbf{x}_k)$
- 6: Define the linear operator $\mathcal{B}_{k+1} : T_{\mathbf{x}_{k+1}} M \rightarrow T_{\mathbf{x}_{k+1}} M$ as follows

$$\mathcal{B}_{k+1}p = \tilde{\mathcal{B}}_k p - \frac{g(s_k, \tilde{\mathcal{B}}_k p)}{g(s_k, \tilde{\mathcal{B}}_k s_k)} \tilde{\mathcal{B}}_k s_k + \frac{g(y_k, p)}{g(y_k, s_k)} y_k, \quad \forall p \in T_{\mathbf{x}_{k+1}} M$$

$$\text{with } \tilde{\mathcal{B}}_k = \mathcal{T}_{\alpha_k\eta_k} \circ \mathcal{B}_k \circ (\mathcal{T}_{\alpha_k\eta_k})^{-1}$$

- 7: **end for**
-

Sherman-Morrison formula

Let A is an invertible matrix. The for all vectors u, v such that $1 + v^T A^{-1} u \neq 0$, one has

$$(A + uv^T)^{-1} = A^{-1} + \frac{A^{-1}uv^T A^{-1}}{1 + v^T A^{-1}u}.$$

Another version of the RBFGS algorithm

Works with the inverse Hessian $\mathcal{H}_k = \mathcal{B}_k^{-1}$ approximation rather than the Hessian approximation B_k . In this case the step 4 in algorithm 2 will be replaced by:

$$\mathcal{H}_{k+1} = \tilde{\mathcal{H}}_k \rho - \frac{g(y_k, \tilde{\mathcal{H}}_k \rho)}{g(y_k, s_k)} s_k - \frac{g(s_k, p_k)}{g(y_k, s_k)} \tilde{\mathcal{H}}_k y_k + \frac{g(s_k, p) g(y_k, \tilde{\mathcal{H}}_k y_k)}{g(y_k, s_k)^2} s_k + \frac{g(s_k, s_k)}{g(y_k, s_k)} \rho$$

with

$$\tilde{\mathcal{H}}_k = \mathcal{T}_{\eta_k} \circ \mathcal{H}_k \circ (\mathcal{T}_{\eta_k})^{-1}$$

Makes it possible to cheaply compute an approximation of the inverse of the Hessian. This may make BFGS advantageous even in the case where we have a cheap exact formula for the Hessian but not for its inverse.

Implementation of RFBFGS in submanifolds of \mathbb{R}^n

Let $x \in M, \xi_x, \eta_x \in T_x M$, define the inclusions:

$$\begin{aligned} i: M &\rightarrow \mathbb{R}^n; x \mapsto i(x) \\ i_x: T_x M &\rightarrow \mathbb{R}^n; \xi_x \mapsto i_x(\xi_x) \end{aligned}$$

use the matrix B_k to represent the linear operator $\mathcal{B}_k: T_{x_k} M \rightarrow T_{x_k} M$.

$$B_k \leftarrow \mathcal{B}_k$$

we have

$$\begin{aligned} i_x(\mathcal{B}_k \xi_x) &= B_k(i_x(\xi_x)) \\ g_x(\xi_x, \eta_x) &= \langle i_x(\xi_x), i_x(\eta_x) \rangle \end{aligned}$$

Compute $\eta_k = -\mathcal{B}_k^{-1} \text{grad } f(x_k)$ for Submanifolds.

Approach 1: Realize \mathcal{B}_k by an n-by-n matrix $B_k^{(n)}$.

Let \mathcal{B}_k be the linear operator $\mathcal{B}_k : T_{x_k} M \rightarrow T_{x_k} M$, $B_k^{(n)} \in \mathbb{R}^{n \times n}$, s.t

$$i_{x_k}(\mathcal{B}_k \eta_k) = B_k^{(n)}(i_{x_k}(\eta_k)), \forall \eta_k \in T_{x_k} M,$$

$$\text{from } \mathcal{B}_k \eta_k = -\text{grad } f(x_k)$$

$$\text{we have } B_k^{(n)}(i_{x_k}(\eta_k)) = -i_{x_k}(\text{grad } f(x_k)).$$

Approach 2: Use bases.

Let $[E_{k,1}, \dots, E_{k,d}] =: \underline{E}_k \in \mathbb{R}^{n \times d}$ be a basis of $T_{x_k} M$. We have

$$\underline{E}_k^+ B_k^{(n)} \underline{E}_k \underline{E}_k^+ i_{x_k}(\eta_k) = -\underline{E}_k^+ i_{x_k}(\text{grad } f(x_k))$$

where $\underline{E}_k^+ = (\underline{E}_k^T \underline{E}_k)^{-1} \underline{E}_k^T$

$$B_k^d = \underline{E}_k^+ B_k^{(n)} \underline{E}_k \in \mathbb{R}^{d \times d}$$

$$B_k^{(d)}(\eta_k)^{(d)} = -(\text{grad } f(x_k))^{(d)}$$

Global convergence of RBFGS

Assumption 1

- (1) The objective function f is twice continuously differentiable
- (2) The level set $\Omega = \{x \in M : f(x) \leq f(x_0)\}$ is convex. In addition, there exists positive constants n and N such that

$$ng(z, z) \leq g(G(x)z, z) \leq Ng(z, z) \text{ for all } z \in M \text{ and } x \in \Omega$$

where $G(x)$ denotes the lifted Hessian.

Theorem

Let B_0 be any symmetric positive definite matrix, and let x_0 be starting point for which assumption 1 is satisfied. Then the sequence x_k generated by algorithm 2 converge to the minimizer of f .

Superlinear convergence of quasi-Newton: generalized Dennis-Moré condition

Let M be a manifold endowed with a C^2 vector transport \mathcal{T} and an associated retraction R . Let F be a C^2 tangent vector field on M . Also let M be endowed with an affine connection ∇ and let $\mathbb{D}F(x)$ denote the linear transformation of $T_x M$ defined by $\mathbb{D}F(x)[\xi_x] = \nabla_{\xi_x} F$ for all tangent vectors ξ_x to M at x . Let $\{\mathcal{B}_k\}$ be a sequence of bounded nonsingular linear transformation of $T_{x_k} M$, where $k = 0, 1, \dots$, $x_{k+1} = R_{x_k}(\eta_k)$, and $\eta_k = -\mathcal{B}_k^{-1} F(x_k)$. Assume that $\mathbb{D}F(x^*)$ is nonsingular, $x_k \neq x^*$, $\forall k$, and $\lim_{k \rightarrow \infty} x_k = x^*$.

Then $\{x_k\}$ converges superlinearly to x^* and $F(x^*) = 0$ if and only if

$$\lim_{k \rightarrow \infty} \frac{\|[\mathcal{B}_k - \mathcal{T}_{\xi_k} \mathbb{D}F(x^*) \mathcal{T}_{\xi_k}^{-1}] \eta_k\|}{\|\eta_k\|} = 0 \quad (17)$$

where $\xi_k \in T_{x^*} M$ is defined by $\xi_k = R_{x^*}^{-1}(x_k)$, i.e. $R_{x^*}(\xi_k) = x_k$.

Superlinear convergence of RBFGS

Assumption 2 The lifted Hessian matrix $\text{Hess} \hat{f}_x$ is Lipschitz-continuous at 0_x uniformly in a neighbourhood of x^* , i.e., there exists $L_* > 0$, $\delta_1 > 0$, and $\delta_2 > 0$ such that, for all $x \in \mathcal{B}_{\delta_1}(x^*)$ and all $\xi \in \mathcal{B}_{\delta_2}(0_x)$, it holds that

$$\|\text{Hess} \hat{f}_x(\xi) - \text{Hess} \hat{f}_x(0_x)\|_x \leq L_* \|\xi\|_x$$

Theorem

Suppose that f is twice continuously differentiable and that the iterates generated by the RBFGS algorithm converge to a nondegenerate minimizer $x^ \in M$ at which Assumption 2 holds. Suppose also that $\sum_{k=1}^{\infty} \|x_k - x^*\| < \infty$ holds. Then x_k converges to x^* at a superlinear rate.*

On the Unit Sphere \mathbb{R}^n

Riemannian metric: $g(\xi, \eta) = \xi^T \eta$

The tangent space at x is:

$$T_x S^{n-1} = \{\xi \in \mathbb{R}^n : x^T \xi = 0\} = \{\xi \in \mathbb{R}^n : x^T \xi + \xi^T x = 0\}$$

Orthogonal projection to tangent space:

$$P_x \xi_x = \xi - x x^T \xi_x$$

Retraction:

$$R_x(\eta_x) = (x + \eta_x) / \|(x + \eta_x)\|, \text{ where } \|\cdot\| \text{ denotes } \langle \cdot, \cdot \rangle^{1/2}$$

Transport on the Unit Sphere \mathbb{R}^n

Parallel Transport of $\xi \in T_x S^{n-1}$ along the geodesic from x in direction $\eta \in T_x S^{n-1}$:

$$P_{\gamma_\eta}^{t \leftarrow 0} \xi = \left(I_n + (\cos(\|\eta\|t) - 1) \frac{\eta\eta^T}{\|\eta\|^2} - \sin(\|\eta\|t) \frac{x\eta^T}{\|\eta\|} \right) \xi;$$

Vector Transport by orthogonal projection:

$$\mathcal{T}_{\eta_x} \xi_x = \left(I - \frac{(x + \eta_x)(x + \eta_x)^T}{\|x + \eta_x\|^2} \right) \xi_x$$

Inverse Vector Transport:

$$(\mathcal{T}_{\eta_x})^{-1}(\xi_{R_x(\eta_x)}) = \left(I - \frac{(x + \eta_x)x^T}{x^T(x + \eta_x)} \right) \xi_{R_x(\eta_x)}$$

On the Unit Sphere

Let $T_{\eta_k}^{(n)}$ be the representation of \mathcal{T}_{η_k}

$$T_{\eta_k}^{(n)} = \left(I - \frac{(x+\eta)(x+\eta)^T}{\|x+\eta\|^2} \right)$$

Approach 1: Realize \mathcal{B}_k by an n-by-n matrix

$$1) \tilde{B}_k^{(n)} = T_{\eta_k}^{(n)} B_k^{(n)} ((T_{\eta_k}^{(n)})^{(n)})^{-1};$$

$$2) B_{k+1}^{(n)} = \tilde{B}_k^{(n)} - \frac{\tilde{B}_k^{(n)} s_k s_k^T \tilde{B}_k^{(n)}}{\langle s_k, \tilde{B}_k^{(n)} s_k \rangle} + \frac{y_k y_k^T}{\langle y_k, s_k \rangle},$$

Approach 2: Use bases

1) Calculate \tilde{B}_k^d though $B_k^{(d)}$:

$$\begin{aligned} \tilde{B}_k^d &= \underline{E}_{k+1}^+ \tilde{B}_k^{(n)} \underline{E}_{k+1}; \\ &= \underline{E}_{k+1}^+ T_{\eta_k}^{(n)} B_k^{(n)} (T_{\eta_k}^{(n)})^{-1} \underline{E}_{k+1} \\ &= \underline{E}_{k+1}^+ T_{\eta_k}^{(n)} \underline{E}_k B_k^{(d)} \underline{E}_k^+ (T_{\eta_k}^{(n)})^{-1} \underline{E}_{k+1} \end{aligned}$$

$$2) B_{k+1}^{(d)} = \tilde{B}_k^{(d)} - \frac{\tilde{B}_k^{(d)} s_k^{(d)} (s_k^{(d)})^T \tilde{B}_k^{(d)}}{\langle s_k^{(d)}, \tilde{B}_k^{(d)} s_k^{(d)} \rangle} + \frac{y_k^{(d)} (y_k^{(d)})^T}{\langle y_k^{(d)}, s_k^{(d)} \rangle},$$

Rayleigh quotient minimization on S^{n-1} Cost function on S^{n-1}

$$f : S^{n-1} \rightarrow \mathbb{R} : x \mapsto x^T A x, A = A^T$$

Cost function embedded in \mathbb{R}^n

$$\bar{f} : \mathbb{R}^n \rightarrow \mathbb{R} : x \mapsto x^T A x, \text{ so that } f = \bar{f} \Big|_{S^{n-1}}$$

$$T_x S^{n-1} = \{\xi \in \mathbb{R}^n : x^T \xi = 0\}, \quad R_x(\xi) = \frac{x + \xi}{\|x + \xi\|}$$

$$D\bar{f}(x)[\zeta] = 2\zeta^T A x \rightarrow \text{grad } \bar{f}(x) = 2Ax$$

$$\text{Projection onto } T_x \mathbb{R}^n : \quad P_x \xi = \xi - x x^T \xi$$

$$\text{Gradient: } \text{grad } f(x) = 2P_x(Ax)$$

Methods Numerical Experiment

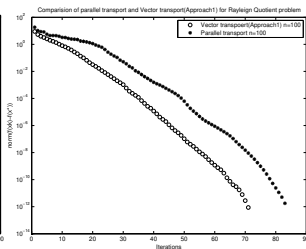
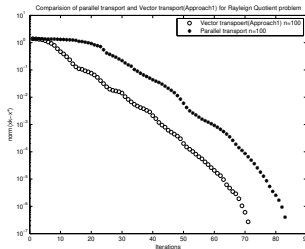
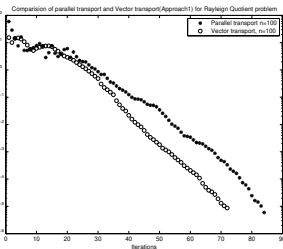
1. Vector transport (approach 1), update $H = B^{-1}, \eta = -H \text{grad } f(x)$
2. Vector transport (approach 2), update $H = B^{-1}, \eta = -H \text{grad } f(x)$
3. Parallel transport, update $H = B^{-1}, \eta = -H \text{grad } f(x)$
4. Vector transport (approach 1), Update L , solve $L_+ L_+^T \eta = -\text{grad } f(x)$ (QR factorization)
5. Riemannian Line Search Newton-CG
6. Riemannian Trust Region with Truncated-CG

Numerical Result for Rayleigh Quotient on S^{n-1}

- ▶ Problem sizes $n = 100$ and $n = 300$ with many different initial points.
- ▶ All versions of RBFSS converge superlinearly to local minimizer.
- ▶ Updating L and B^{-1} combined with Vector transport display similar convergence rates.
- ▶ Vector transport Approach 1 and Approach 2 display the same convergence rate, but Approach 2 takes more time due to complexity of each step.
- ▶ The updated B^{-1} of Approach 2 and Parallel transport has better conditioning, i.e. more positive definite.
- ▶ Vector transport versions converge faster than Parallel transport. On S^{n-1} , they have similar computational cost.
- ▶ Newton–CG version converges slightly more quickly than the Vector transport versions.

Rayleigh quotient on S^{n-1}

Vector transport has better convergence rate than Parallel transport



Rayleigh quotient on S^{n-1}

Table: Comparison of Vector transport vs. Parallel translation for Rayleigh quotient Problem

Case	Vector trans. (n=100)	Vector trans. (n=300)	Parallel trans. (n=100)	Parallel trans. (n=300)
Time	0.22	4.06	0.46	5.49
Iteration	71	97	84	95

Table: Vector transport approach1 vs. approach2 for Rayleigh quotient problem

Case	approach 1 (n=100)	approach 1 (n=300)	approach 2 (n=100)	approach 2 (n=300)
Time	0.22	4.06	2.2	33.6
Iteration	71	97	71	97

Other vector transports on S^{n-1}

- ▶ NI: nonisometric vector transport by orthogonal projection onto the new tangent space (see above)
- ▶ CB: a vector transport relying on the canonical bases between the current and next subspaces
- ▶ CBE: a mathematically equivalent but computationally efficient form of CB
- ▶ QR: the basis in the new subspace is obtained by orthogonal projection of the previous basis followed by Gram-Schmidt.

Rayleigh quotient, $n = 300$

	NI	CB	CBE	QR
Time (sec.)	4.0	20	4.7	15.8
Iteration	97	92	92	97

On the Manifold $S^{n-1} \times \dots \times S^{n-1}$

$$\begin{aligned}
 X &= [x_1, x_2, \dots, x_N] \in S^{n-1} \times \dots \times S^{n-1} \\
 x_i^T x_i &= 1, \text{ for } i = 1 \text{ to } N
 \end{aligned}$$

Riemannian metric:

$$\ll Z, W \gg_X = \langle z_1, w_1 \rangle_{x_1} + \dots + \langle z_N, w_N \rangle_{x_N} = \text{tr}(Z^T W), Z, W \in T_X \mathcal{M}$$

Tangent space at x :

$$T_x \mathcal{M} = \{Z = [z_1, \dots, z_N] \in \mathbb{R}^{n \times N} \mid x_1^T z_1 = x_2^T z_2 = \dots = x_N^T z_N = 0\}$$

Orthogonal projection to tangent space:

$$P_X W = [(I - x_1 x_1^T) w_1, \dots, (I - x_N x_N^T) w_N] \text{ projects } W \in \mathbb{R}^{n \times N} \text{ to } T_x \mathcal{M}$$

Retraction:

$$R_X(Z) = \left[\frac{x_1 + z_1}{\|x_1 + z_1\|}, \dots, \frac{x_N + z_N}{\|x_N + z_N\|} \right]$$

Transport on $S^{n-1} \times \dots \times S^{n-1}$

Parallel and vector transport (and their inverses) of

$$\xi_X = [\xi_1, \xi_2, \dots, \xi_N] \in T_x \mathcal{M}$$

defined by directions

$$\eta_X = [\eta_1, \eta_2, \dots, \eta_N] \in T_x \mathcal{M}$$

simply apply the corresponding transport mechanisms from S^{n-1} componentwise.

Thomson Problem on $S^{n-1} \times \dots \times S^{n-1}$

$$X = [x_1, x_2, \dots, x_N] \in \mathcal{M}, x_i^T x_i = 1, \text{ for } i = 1 \text{ to } N$$

$$f : [x_1, x_2, \dots, x_N] \mapsto \sum_{\substack{i,j=1 \\ i \neq j}}^N \frac{1}{\|x_i - x_j\|^2}$$

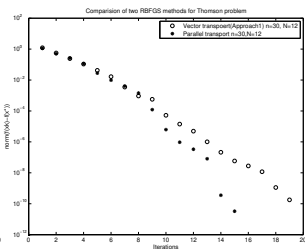
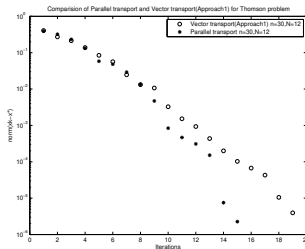
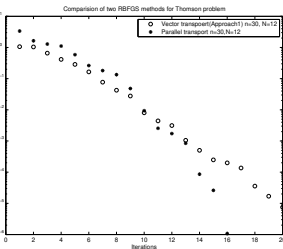
$$\text{grad } f(X) = \left[(I - x_1 x_1^T) \sum_{j=2}^N \frac{1}{(1 - x_1^T x_j)^2} x_j, \dots, (I - x_N x_N^T) \sum_{j=1}^{N-1} \frac{1}{(1 - x_N^T x_j)^2} x_j \right]$$

Methods Numerical Experiment

1. Vector transport (approach 1), update $H = B^{-1}, \eta = -H \text{grad } f(x)$
2. Vector transport (approach 2), update $H = B^{-1}, \eta = -H \text{grad } f(x)$
3. Parallel transport (approach 1), update $H = B^{-1}, \eta = -H \text{grad } f(x)$
4. Vector transport (approach 1), Update L , solve $L_+ L_+^T \eta = -\text{grad } f(x)$ (QR factorization)
5. Riemannian Trust Region with Truncated-CG

Numerical Result for Thomson Problem

- ▶ Problem sizes $(n, N) = (30, 12)$ and $(n, N) = (50, 20)$ with many different initial points.
- ▶ All versions of RBFGS converge superlinearly to local minimizer.
- ▶ Updating L and B^{-1} combined with Vector transport display similar convergence rates.
- ▶ Vector transport Approach 1 and Approach 2 display the same convergence rate, but Approach 2 takes more time due to complexity of each step.
- ▶ The updated B^{-1} of Approach 2 and Parallel transport has better conditioning, i.e. more positive definite.
- ▶ Parallel transport converge slightly faster than Vector transport versions .

Update of B^{-1} , Parallel and Vector Transport

Update of B^{-1} , Parallel and Vector Transport

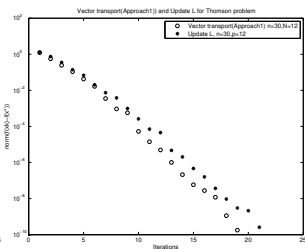
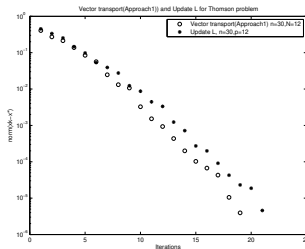
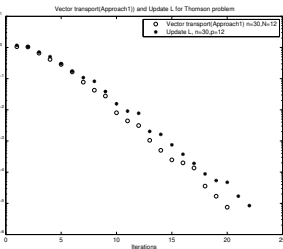
Table: Vector transport (approach 1) vs. Parallel transport for Thomson problem

Case	Vector trans. (n=30, N=12)	Vector trans. (n=50, N=20)	Parallel trans. (n=30, N=12)	Parallel trans. (n=50, N=20)
Time	3.9	60	3.4	47.6
Iteration	20	24	16	19

Table: Vector transport (approach 1) vs. Parallel transport (approach 1) for Thomson problem

Case	approach 1 (n=30, N=12)	approach 1 (n=50, N=20)	approach 2 (n=30, N=12)	approach 2 (n=50, N=20)
Time	3.9	60	13	252
Iteration	20	24	20	24

Update L and Update of B^{-1} for Thomson Problem



Update of B^{-1} and Riemannian Trust Region Method

- ▶ Total inner iteration count of RTR is larger than iteration count of R BFGS
- ▶ RTR inner iteration and RBFGS iteration similar complexity

Update of B^{-1} and Riemannian Trust Region Method

Table: RBFGS (Vector transport, approach 1) vs. RTR for Rayleigh Quotient problem

Case	RBFGS ($n=30, N=12$)	RBFGS ($n=50, N=20$)	RTR ($n=30, N=12$)	RTR ($n=50, N=20$)
Iteration	20	24	30	36

Compact Stiefel Manifold $St(p, n)$

View $St(p, n)$ as a Riemannian submanifold of the Euclidean space $\mathbb{R}^{n \times p}$

Riemannian metric: $g(\xi, \eta) = \text{tr}(\xi^T \eta)$

The tangent space at X is:

$$T_X St(p, n) = \{Z \in \mathbb{R}^{n \times p} : X^T Z + Z^T X = 0\}.$$

Orthogonal projection to tangent space is :

$$P_X \xi_X = (I - XX^T) \xi_X + X \text{skew}(X^T \xi_X)$$

Retraction:

$$R_X(\eta_X) = \text{qf}(X + \eta_X)$$

where $\text{qf}(A) = Q \in \mathbb{R}_*^{n \times p}$, where $A = QR$

Parallel Transport On Stiefel Manifold

Let $Y^T Y = I_p$ and $A = Y^T H$ is skew-symmetric. The geodesic from Y in direction H :

$$\gamma_H(t) = YM(t) + QN(t),$$

Q and R : the compact QR decomposition of $(I - YY^T)H$
 $M(t)$ and $N(t)$ given by:

$$\begin{pmatrix} M(t) \\ N(t) \end{pmatrix} = \exp\left(t \begin{pmatrix} A & -R^T \\ R & 0 \end{pmatrix}\right) \begin{pmatrix} I_p \\ 0 \end{pmatrix}$$

The parallel transport of H along the geodesic from Y in direction H :

$$P_{\gamma_H}^{t \leftarrow 0} H = HM(t) - YR^T N(t)$$

Parallel Transport On Stiefel Manifold

The parallel transport of $\xi \in H$ along the geodesic, $\gamma(t)$, from Y in direction H :

$$\begin{aligned}w(t) &= P_{\gamma}^{t \leftarrow 0} \xi \\w'(t) &= -\frac{1}{2} \gamma(t) (\gamma'(t)^T w(t) + w(t)^T \gamma'(t)), w(0) = \xi\end{aligned}$$

In practice, the ODE is solved discretely.

Vector Transport on $St(p, n)$ Approach 1

$$\mathcal{T}_{\eta_X} \xi_X = (I - YY^T)\xi_X + Y \text{skew}(Y^T \xi_X), \text{ where } Y := R_X(\eta_X)$$

$$(\mathcal{T}_{\eta_X})^{-1} \xi_Y = \xi_Y + YS, \text{ where } Y := R_X(\eta_X)$$

S is symmetric matrix such that $X^T(\xi_Y + YS)$ is skew-symmetric.

Vector Transport on $St(p, n)$ Approach2

- ▶ Find d independent tangent vectors $E_{k,1}, E_{k,2}, \dots, E_{k,d} \in T_{X_k}$;
- ▶ Vector transport each $E_{ki}, i = 1, 2, \dots, d$ to $T_{X_{k+1}}$,

$$\underline{E}_{k+1} = [T_{\eta_k}^{(np)} E_{k,1} \quad T_{\eta_k}^{(np)} E_{k,2} \quad \dots \quad T_{\eta_k}^{(np)} E_{k,d}]$$
- ▶ Calculate $\tilde{B}_k^{(np)} = T_{\eta_k}^{(np)} B_k^{(np)} (T_{\eta_k}^{(np)})^{-1}$:

$$\tilde{B}_k^{(np)} \underline{E}_{k+1} = [T_{\eta_k}^{(np)} (B_k^{(np)} E_{k,1}) \quad T_{\eta_k}^{(np)} (B_k^{(np)} E_{k,2}) \quad \dots \quad T_{\eta_k}^{(np)} (B_k^{(np)} E_{k,d})],$$

$$\tilde{B}_k^{(np)} = [T_{\eta_k}^{(np)} (B_k^{(np)} E_{k,1}) \quad T_{\eta_k}^{(np)} (B_k^{(np)} E_{k,2}) \quad \dots \quad T_{\eta_k}^{(np)} (B_k^{(np)} E_{k,d})] \underline{E}_{k+1}^+.$$

- ▶ Compute the RFBFGS update

$$B_{k+1}^{(np)} = \tilde{B}_k^{(np)} - \frac{\tilde{B}_k^{(np)} s_k^{(np)} s_k^{(np)T} \tilde{B}_k^{(np)}}{\langle s_k^{(np)}, \tilde{B}_k^{(np)} s_k^{(np)} \rangle} + \frac{y_k^{(np)} y_k^{(np)T}}{\langle y_k^{(np)}, s_k^{(np)} \rangle}, \text{ and set}$$

$$\eta_{k+1} = \text{unvec}\{(-B_{k+1}^{(np)})^{-1} \text{vec}\{\text{grad } f(X_k)\}\}.$$

A Procrustes Problem on $\text{St}(p, n)$

Cost function on $\text{St}(p, n)$

$$f : \text{St}(p, n) \rightarrow \mathbb{R} : X \rightarrow \|AX - XB\|_F$$

where $A: n \times n$ matrix, $B: p \times p$ matrix, $X^T X = I_p$

Cost function embedded in $\mathbb{R}^{n \times p}$

$$\bar{f} : \mathbb{R}^{n \times p} \rightarrow \mathbb{R} : X \rightarrow \|AX - XB\|_F, \quad \text{with } f = \bar{f}|_{\text{St}(p, n)}$$

$$T_X \text{St}(p, n) = \{Z \in \mathbb{R}^{n \times p} : X^T Z + Z^T X = 0\}$$

$$D\bar{f}(X)[Z] = \frac{\text{tr}(Z^T Q)}{\bar{f}(X)}, \quad \text{where } Q = A^T A X - A^T X B - B^T A X + B^T X B,$$

Projection onto $T_x \mathbb{R}^n$:

$$P_X Z = (I - X X^T) Z + X \text{skew}(X^T Z)$$

$$\text{Gradient: } \quad \text{grad } f(X) = P_X \text{grad } \bar{f}(x)$$

Methods Numerical Experiment

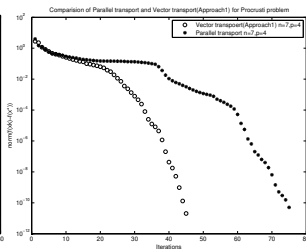
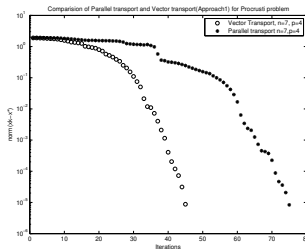
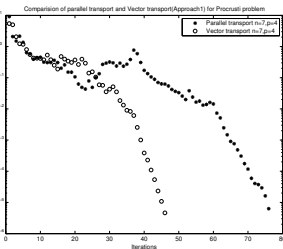
1. Vector transport (approach 1), update $H = B^{-1}, \eta = -H \text{grad } f(x)$
2. Vector transport (approach 2), update $H = B^{-1}, \eta = -H \text{grad } f(x)$
3. Parallel transport, update $H = B^{-1}, \eta = -H \text{grad } f(x)$
4. Vector transport (approach 1), Update L , solve $L_+ L_+^T \eta = -\text{grad } f(x)$ (QR factorization)
5. Riemannian Line Search Newton-CG
6. Riemannian Trust Region with Truncated-CG

Numerical Result for Procrustes on $\text{St}(p, n)$

- ▶ Problem sizes $(n, p) = (7, 4)$ and $(n, p) = (12, 7)$ with many different initial points.
- ▶ All versions of RBFGS converge superlinearly to local minimizer.
- ▶ Updating L and B^{-1} combined with Vector transport display B^{-1} is slightly faster converging.
- ▶ Vector transport Approach 1 and Approach 2 display the same convergence rate, but Approach 2 takes more time due to complexity of each step.
- ▶ The updated B^{-1} of Approach 2 and Parallel transport has better conditioning, i.e. more positive definite.
- ▶ Vector transport versions converge noticeably faster than Parallel transport. This depends on numerical evaluation of ODE for Parallel transport.
- ▶ Newton–CG version has convergence problems compared to the Vector transport RBFGS versions.

Procrustes Problem on $St(p, n)$

Vector transport has better convergence rate than Parallel transport



Procrustes Problem on $\text{St}(p, n)$ Table: B^{-1} update w/ Vector transport (approach 1) vs. Parallel transport

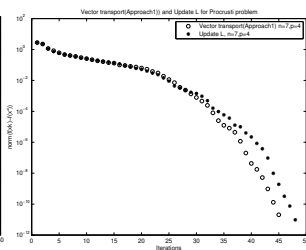
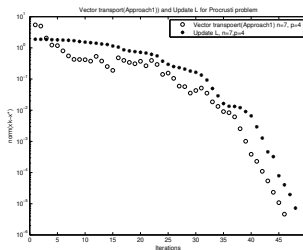
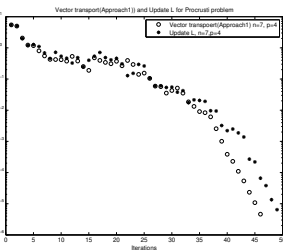
Case	Vector trans. ($n=7, p=4$)	Vector trans. ($n=12, p=7$)	Parallel trans. ($n=7, p=4$)	Parallel trans. ($n=12, p=7$)
Time	4.1	45	81	781
Iteration	46	82	67	174

Table: Vector transport approach1 vs. approach2 for Procrustes problem

Case	approach 1 ($n=7, p=4$)	approach 1 ($n=12, p=7$)	approach 2 ($n=7, p=4$)	approach 2 ($n=12, p=7$)
Time	4.1	46	7.5	95
Iteration	46	82	48	86

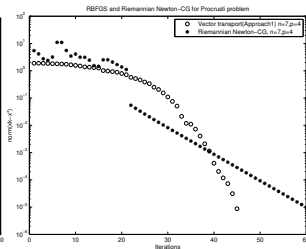
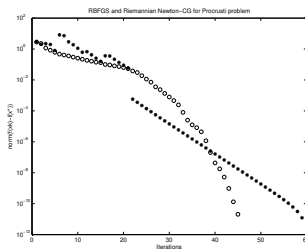
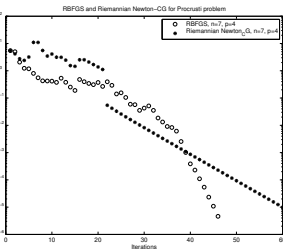
Update of L and Update of B^{-1}

- ▶ Both $O(n^2)$ operations per step and use Vector transport with Approach 1.
- ▶ Similar convergence behavior



Update of B^{-1} and Riemannian Line Search Newton–CG

- The Convergence of RBFGS is superlinear, while Newton–CG is linear since no forcing function used in CG convergence check.



Update of B^{-1} and Riemannian Trust Region Method

- ▶ Total inner iteration count of RTR is larger than iteration count of RBFGS
- ▶ RTR inner iteration and RBFGS iteration similar complexity

Comparison of RBFGS with Riemannian Trust Region Method

Table: RBFGS (Vector transport, approach 1) vs. RTR for Procrustes problem

Case	RBFGS ($n=7, p=4$)	RBFGS ($n=12, p=7$)	RTR ($n=7, p=4$)	RTR ($n=12, p=7$)
Iteration	47	86	115	357

A (questionable) historical overview

	In \mathbb{R}^n	On Riemannian manifolds	
		using classical objects	using novel objects
Steepest descent	1966 (Armijo backtracking)	1972 (Luenberger)	1986–2008 ?
Newton	1740 (Simpson)	1993 (Smith)	2002 (Adler et al.)
Conjugate Grad	1964 (Fletcher–Reeves)	1993 (Smith)	2008 (PAA, Mahony, Sepulchre) ?
Trust regions	1985 (name created by Celis, Dennis, Tapia)	2007 (PAA, Baker, Gallivan)	2007 (PAA, Baker, Gallivan)
BFGS	1970 (B-F-G-S)	1982 (Gabay)	Now!

News flashes

Manopt — a Matlab toolbox for optimization on manifolds

Latest commit: 19th Jun 2019

Context:

- ▶ When it was first released on 3rd Jan 2013, Manopt was arguably the first toolbox based on a separation of the manifolds, the solvers, and the problem description.
- ▶ Developed by Nicolas Boumal (Princeton University) and Bamdev Mishra (Amazon Development Centre India).
- ▶ Originated at the Universities of Louvain and Liège.
- ▶ See <https://www.manopt.org> for download, tutorial, forum...
- ▶ See <http://jmlr.org/papers/v15/boumal14a.html> for a paper presenting the toolbox.

Manopt — a Matlab toolbox for optimization on manifolds

Latest commit: 19th Jun 2019

Context:

- ▶ When it was first released on 3rd Jan 2013, Manopt was arguably the first toolbox based on a separation of the manifolds, the solvers, and the problem description.
- ▶ Developed by Nicolas Boumal (Princeton University) and Bamdev Mishra (Amazon Development Centre India).
- ▶ Originated at the Universities of Louvain and Liège.
- ▶ See <https://www.manopt.org> for download, tutorial, forum...
- ▶ See <http://jmlr.org/papers/v15/boumal14a.html> for a paper presenting the toolbox.

Ongoing and future work:

- ▶ Implement solvers for nonsmooth cost functions.
- ▶ Implement stochastic gradient methods on manifolds.
- ▶ Implement automatic differentiation features.
- ▶ ...

Pymanopt — a Python toolbox for optimization on manifolds

Latest commit: 3rd May 2019

- ▶ Pymanopt is a Python toolbox for optimization on manifolds with support for automatic differentiation
- ▶ Developers: Jamie Townsend, Niklas Koep, Sebastian Weichwald
- ▶ Announced on 25th Nov 2015
- ▶ Paper: 17(137):1–5, 2016,
<http://jmlr.org/papers/v17/16-177.html>

ROPTLIB — C++ Riemannian Manifold Optimization Library

Latest commit: 25th Mar 2019

- ▶ ROPTLIB is a C++ object-oriented library for optimization on Riemannian manifolds
- ▶ Developer: Wen Huang (Rice University)
- ▶ First version: 20th Feb 2015
- ▶ Interface with Matlab and Julia
- ▶ Paper:
<http://www.math.fsu.edu/~whuang2/papers/ROPTLIB.htm>

ROPTLIB — C++ Riemannian Manifold Optimization Library

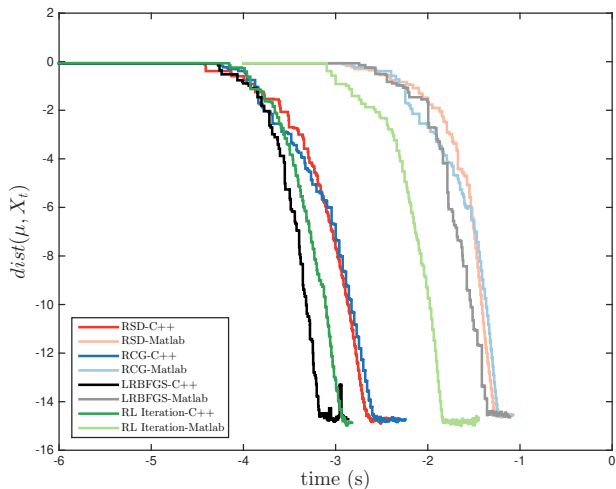
Latest commit: 25th Mar 2019

- ▶ ROPTLIB is a C++ object-oriented library for optimization on Riemannian manifolds
- ▶ Developer: Wen Huang (Rice University)
- ▶ First version: 20th Feb 2015
- ▶ Interface with Matlab and Julia
- ▶ Paper:
<http://www.math.fsu.edu/~whuang2/papers/ROPTLIB.htm>

As an illustration, next comes an experiment where the (Karcher) mean (in the sense of the affine-invariant metric) of three 3×3 positive-definite matrices is computed. Figure made by Xinru Yuan (Florida State University).

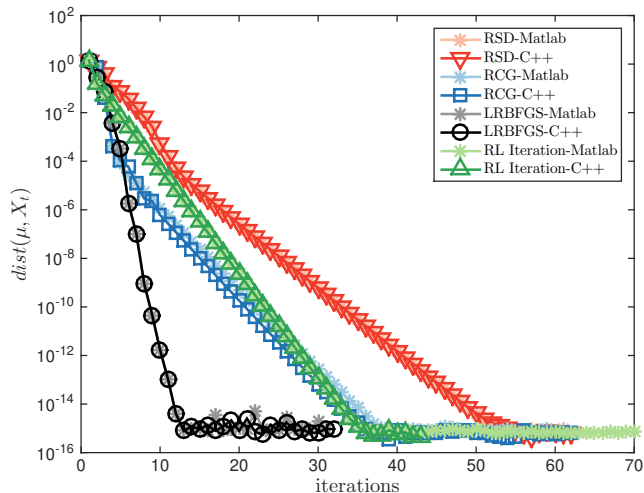
ROPTLIB — C++ Riemannian Manifold Optimization Library I

Latest commit: 25th Mar 2019



ROPTLIB — C++ Riemannian Manifold Optimization Library II

Latest commit: 25th Mar 2019



“Composite” optimization on Stiefel and Grassmann

[GLCY18] - eprint available on 30th Sep 2016

- ▶ A well-known “smooth–nonsmooth” composite optimization problem:

$$\min_{x \in \mathbb{R}^n} \varphi(x) \equiv f(x) + g(x),$$

where f is a smooth (possibly nonconvex) function, while g is a proper, closed, convex (possibly nonsmooth) function with cheaply computable proximal mapping; see, e.g., [STP16].

“Composite” optimization on Stiefel and Grassmann

[GLCY18] - eprint available on 30th Sep 2016

- ▶ A well-known “smooth–nonsmooth” composite optimization problem:

$$\min_{x \in \mathbb{R}^n} \varphi(x) \equiv f(x) + g(x),$$

where f is a smooth (possibly nonconvex) function, while g is a proper, closed, convex (possibly nonsmooth) function with cheaply computable proximal mapping; see, e.g., [STP16].

- ▶ A new type of “Grassmann–Stiefel” composite optimization problem:

$$\min_{\substack{X \in \mathbb{R}^{n \times p} \\ X^T X = I_p}} \varphi(X) \equiv h(X) + \text{trace}(G^T X),$$

where $G \in \mathbb{R}^{n \times p}$ and h is orthogonally invariant, i.e., $h(XQ) = h(X)$ for all orthogonal matrix Q ; see [GLCY18].

“Composite” optimization on Stiefel and Grassmann

[GLCY18] - eprint available on 30th Sep 2016

- ▶ A well-known “smooth–nonsmooth” composite optimization problem:

$$\min_{x \in \mathbb{R}^n} \varphi(x) \equiv f(x) + g(x),$$

where f is a smooth (possibly nonconvex) function, while g is a proper, closed, convex (possibly nonsmooth) function with cheaply computable proximal mapping; see, e.g., [STP16].

- ▶ A new type of “Grassmann–Stiefel” composite optimization problem:

$$\min_{\substack{X \in \mathbb{R}^{n \times p} \\ X^T X = I_p}} \varphi(X) \equiv h(X) + \text{trace}(G^T X),$$

where $G \in \mathbb{R}^{n \times p}$ and h is orthogonally invariant, i.e., $h(XQ) = h(X)$ for all orthogonal matrix Q ; see [GLCY18].

- ▶ In other words, $\text{trace}(G^T X)$ is an easy function on the Stiefel manifold and h is a (general) function on the (smaller) Grassmann manifold.

Global rates of convergence for nonconvex optimization on manifolds

[BAC16] - eprint available on 25th May 2016

- ▶ Standard results on the rate of convergence are local. For example, Newton's method converges quadratically,

$$\|x_{k+1} - x^*\| \leq C \|x_k - x^*\|^2,$$

in a neighborhood of a nondegenerate local minimizer x^* .

Global rates of convergence for nonconvex optimization on manifolds

[BAC16] - eprint available on 25th May 2016

- ▶ Standard results on the rate of convergence are local. For example, Newton's method converges quadratically,

$$\|x_{k+1} - x^*\| \leq C \|x_k - x^*\|^2,$$

in a neighborhood of a nondegenerate local minimizer x^* .

- ▶ Global results are known in \mathbb{R}^n . For example, for a Lipschitz differentiable objective function, an upper bound in $\mathcal{O}(1/\varepsilon^2)$ is known on the number of iterations needed by a gradient descent method to compute a point x where $\|\text{grad } f(x)\| \leq \varepsilon$.

Global rates of convergence for nonconvex optimization on manifolds [BAC16] - eprint available on 25th May 2016

- ▶ Standard results on the rate of convergence are local. For example, Newton's method converges quadratically,

$$\|x_{k+1} - x^*\| \leq C \|x_k - x^*\|^2,$$

in a neighborhood of a nondegenerate local minimizer x^* .

- ▶ Global results are known in \mathbb{R}^n . For example, for a Lipschitz differentiable objective function, an upper bound in $\mathcal{O}(1/\varepsilon^2)$ is known on the number of iterations needed by a gradient descent method to compute a point x where $\|\text{grad } f(x)\| \leq \varepsilon$.
- ▶ A similar result is presented in [BAC16] for steepest-descent on manifolds.

Global rates of convergence for nonconvex optimization on manifolds

[BAC16] - eprint available on 25th May 2016

- ▶ Standard results on the rate of convergence are local. For example, Newton's method converges quadratically,

$$\|x_{k+1} - x^*\| \leq C \|x_k - x^*\|^2,$$

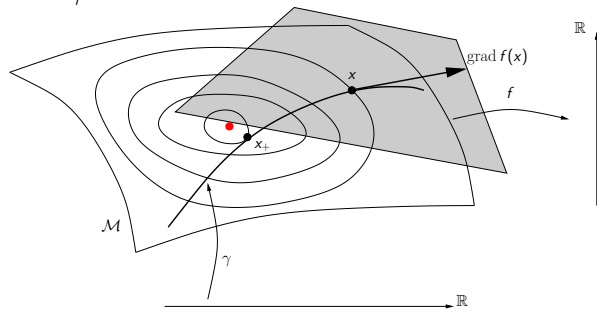
in a neighborhood of a nondegenerate local minimizer x^* .

- ▶ Global results are known in \mathbb{R}^n . For example, for a Lipschitz differentiable objective function, an upper bound in $\mathcal{O}(1/\varepsilon^2)$ is known on the number of iterations needed by a gradient descent method to compute a point x where $\|\text{grad } f(x)\| \leq \varepsilon$.
- ▶ A similar result is presented in [BAC16] for steepest-descent on manifolds.
- ▶ See also [ZS16] that treats geodesically convex problems on Hadamard manifolds.

Barzilai–Borwein step size on manifolds

[IP18] - eprint available on 23rd Dec 2015

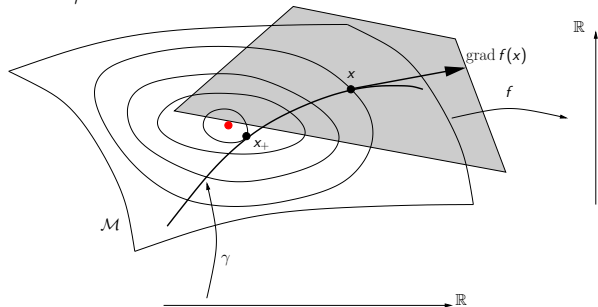
- ▶ On a manifold, the restriction of f to a curve γ is a classical function $f \circ \gamma : \mathbb{R} \rightarrow \mathbb{R}$.



Barzilai–Borwein step size on manifolds

[IP18] - eprint available on 23rd Dec 2015

- ▶ On a manifold, the restriction of f to a curve γ is a classical function $f \circ \gamma : \mathbb{R} \rightarrow \mathbb{R}$.

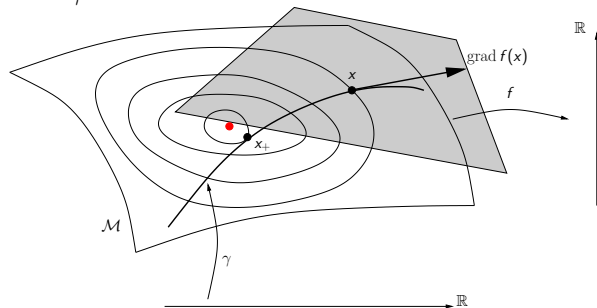


- ▶ In \mathbb{R}^n , the BB stepsize is $\alpha_{k+1}^{\text{BB}} = \frac{s_k^T s_k}{s_k^T y_k}$ where $s_k := x_{k+1} - x_k$ and $y_k := \nabla f(x_{k+1}) - \nabla f(x_k)$.

Barzilai–Borwein step size on manifolds

[IP18] - eprint available on 23rd Dec 2015

- ▶ On a manifold, the restriction of f to a curve γ is a classical function $f \circ \gamma : \mathbb{R} \rightarrow \mathbb{R}$.

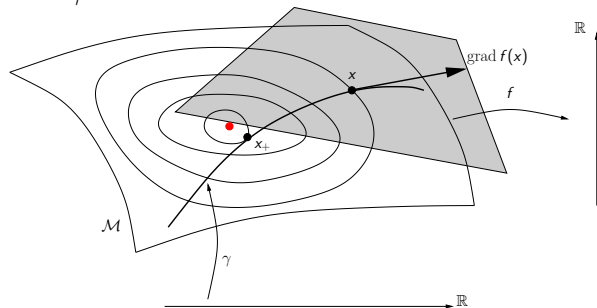


- ▶ In \mathbb{R}^n , the BB stepsize is $\alpha_{k+1}^{\text{BB}} = \frac{s_k^T s_k}{s_k^T y_k}$ where $s_k := x_{k+1} - x_k$ and $y_k := \nabla f(x_{k+1}) - \nabla f(x_k)$.
- ▶ How to generalize BB to manifolds?

Barzilai–Borwein step size on manifolds

[IP18] - eprint available on 23rd Dec 2015

- ▶ On a manifold, the restriction of f to a curve γ is a classical function $f \circ \gamma : \mathbb{R} \rightarrow \mathbb{R}$.

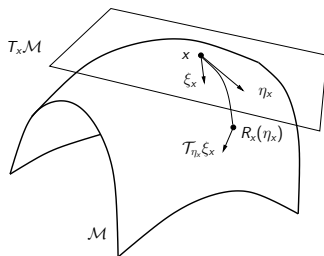


- ▶ In \mathbb{R}^n , the BB stepsize is $\alpha_{k+1}^{\text{BB}} = \frac{s_k^T s_k}{s_k^T y_k}$ where $s_k := x_{k+1} - x_k$ and $y_k := \nabla f(x_{k+1}) - \nabla f(x_k)$.
- ▶ How to generalize BB to manifolds?
- ▶ See [IP15]. It often gives excellent results in practice.

Isometric vector transport on the Stiefel manifold

[Zhu16] - eprint available on 4th Sep 2016

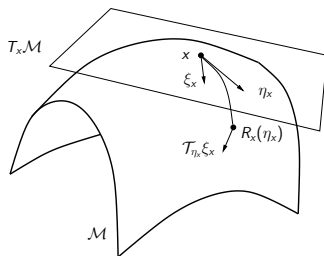
- ▶ In many optimization methods, one need to move tangent vectors.
- ▶ In the important BFGS method on manifolds, the motion (called *vector transport*) needs to be isometric.



Isometric vector transport on the Stiefel manifold

[Zhu16] - eprint available on 4th Sep 2016

- ▶ In many optimization methods, one need to move tangent vectors.
- ▶ In the important BFGS method on manifolds, the motion (called *vector transport*) needs to be isometric.



- ▶ Zhu [Zhu16, §3.3] has proposed an isometric vector transport on the Stiefel manifold with an elegant compact formula:

$$\mathcal{T}_{\eta_X} \xi_X = \left(I - \frac{1}{2} W_{\eta_X}\right)^{-1} \left(I + \frac{1}{2} W_{\eta_X}\right) \xi_X,$$

where $W_{\eta_X} := P_X \eta_X X^T - X \eta_X^T P_X$ and $P_X = I - \frac{1}{2} X X^T$.

An optimization-oriented introduction to optimization on manifolds

[HLWY19] - eprint available on 13th Jun 2019

Applications mentioned in:

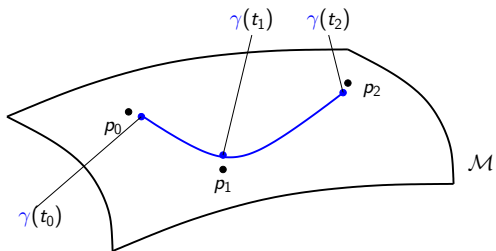
- ▶ P-harmonic flow (image analysis)
- ▶ Max cut
- ▶ Nearest low-rank correlation matrix
- ▶ Phase retrieval
- ▶ Bose-Einstein condensates
- ▶ Cryo-EM
- ▶ Linear and nonlinear eigenvalue problems
- ▶ Batch normalization in deep learning
- ▶ Sparse PCA
- ▶ Low-rank matrix completion
- ▶ ...

Data fitting on manifolds

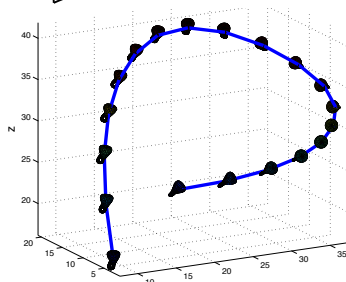
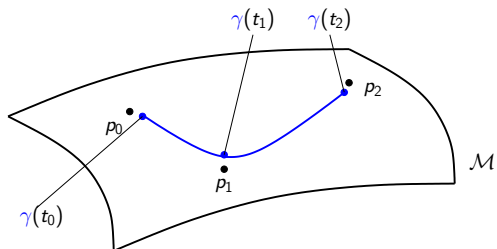
Collaborations

- ▶ Nonparametric curve fitting on manifolds:
 - ▶ Chafik Samir, PAA, Anuj Srivastava, Eric Klassen, *A gradient-descent method for curve fitting on Riemannian manifolds*, Foundations of Computational Mathematics, 12(1), pp. 49-73, 2012.
 - ▶ Nicolas Boumal, PAA, *Discrete regression methods on the cone of positive-definite matrices*, ICASSP 2011.
 - ▶ Nicolas Boumal, PAA, *A discrete regression method on manifolds and its application to data on $SO(n)$* , IFAC World Congress 2011.
- ▶ Parametric curve fitting on manifolds:
 - ▶ C. Samir, P. Van Dooren, D. Laurent, K. A. Gallivan, PAA, *Elastic morphing of 2D and 3D objects on a shape manifold*, Lecture Notes in Computer Science, Volume 5627/2009, pp. 563-572, 2009.
 - ▶ Antoine Arnould, Pierre-Yves Gousenbourger, Chafik Samir, PAA, *Fitting Smooth Paths on Riemannian manifolds: Endometrial Surface Reconstruction and Preoperative MRI-Based Navigation*, GSI'15.
 - ▶ PAA, Pierre-Yves Gousenbourger, Paul Striowski, Benedikt Wirth, *Differentiable piecewise-Bézier surfaces on Riemannian manifolds*, SIAM Journal on Imaging Sciences, 2016
 - ▶ Pierre-Yves Gousenbourger, Estelle M. Massart, Antoni Musolas, P.-A. Absil, Julien M. Hendrickx, Laurent Jacques, Youssef Marzouk, *Piecewise-Bézier C^1 smoothing on manifolds with application to wind field estimation*, submitted, 2016.

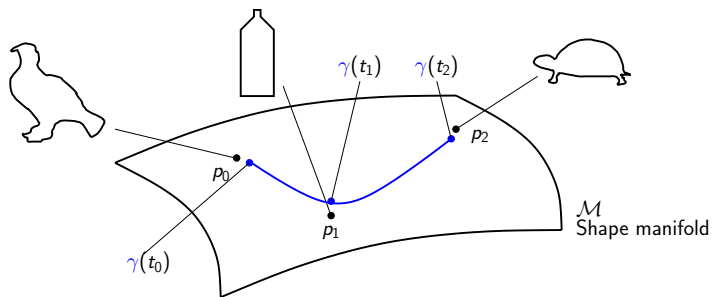
Curve fitting on manifolds



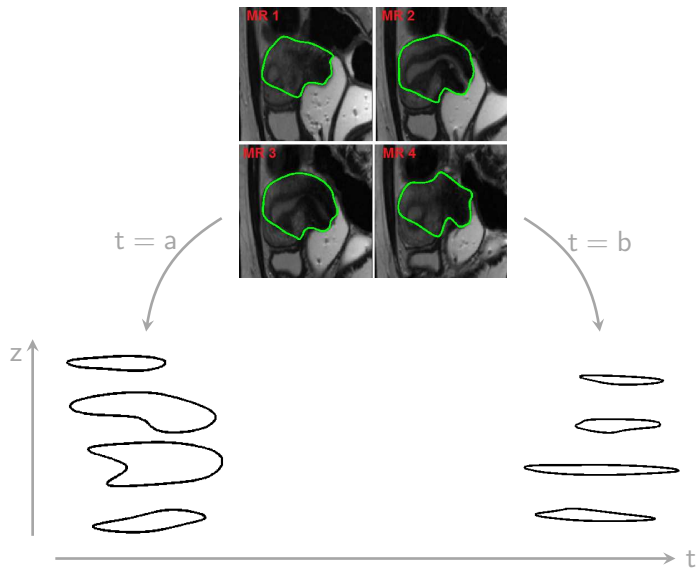
Curve fitting on manifolds: Medical application



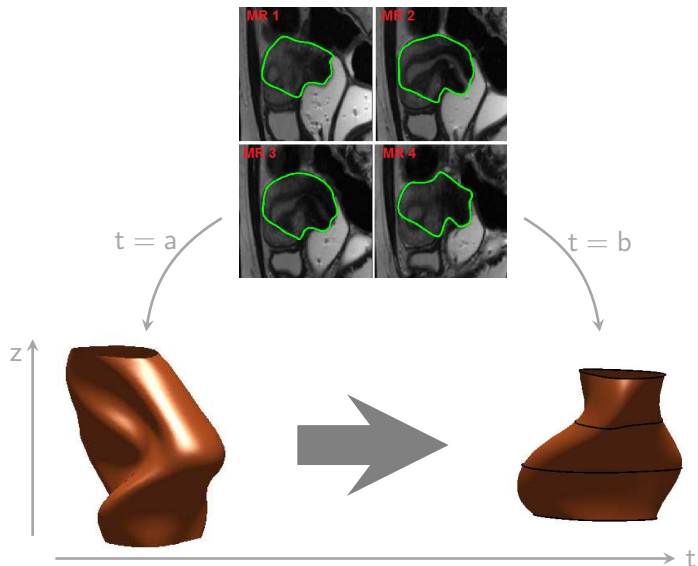
Curve fitting on manifolds: Application to morphing



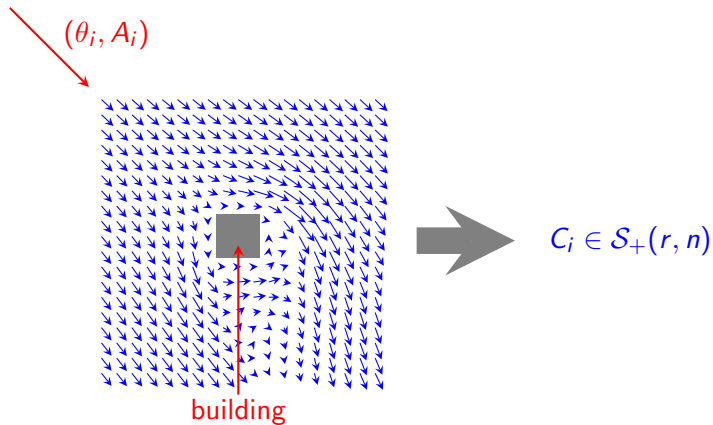
Curve fitting on manifolds: Application to morphing



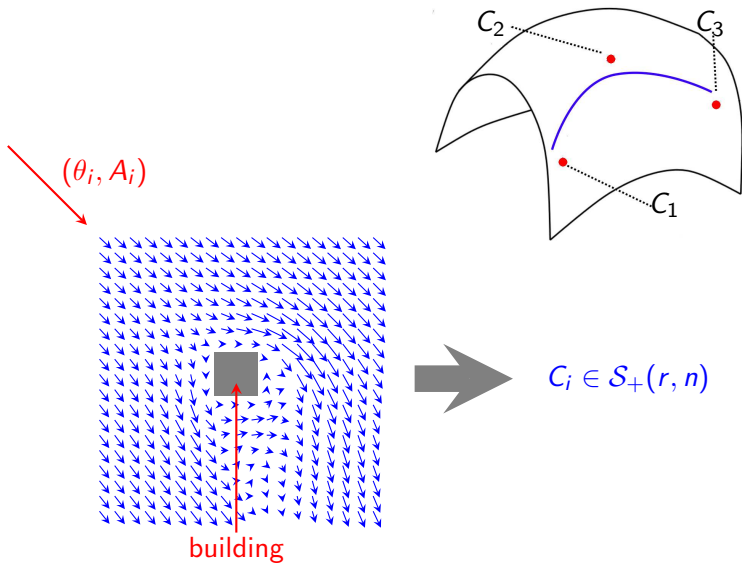
Curve fitting on manifolds: Application to morphing



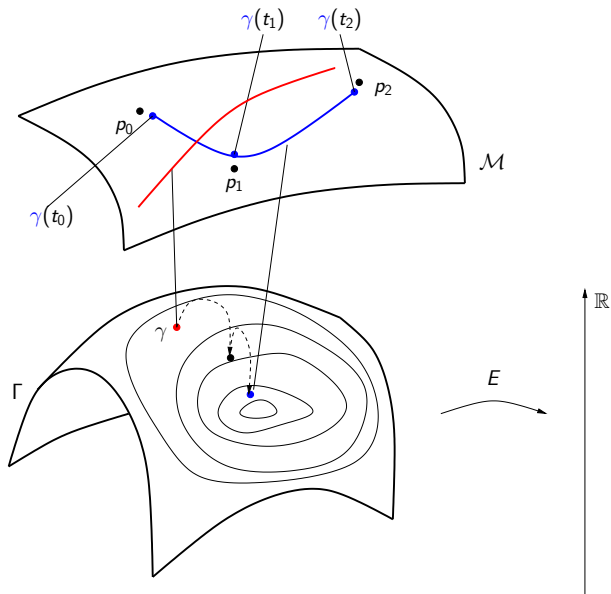
Curve fitting on manifolds: Wind field estimation



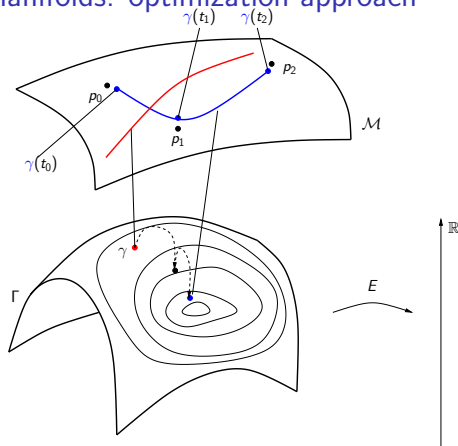
Curve fitting on manifolds: Wind field estimation



Curve fitting on manifolds: optimization approach



Curve fitting on manifolds: optimization approach



Goal: find the curve $\gamma : [0, 1] \mapsto \mathcal{M}$ that minimizes

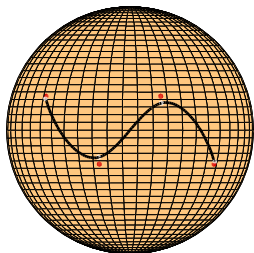
$$E_2 : \Gamma_2 \rightarrow \mathbb{R} : E_2(\gamma) = \frac{1}{2} \sum_{i=0}^N d^2(\gamma(t_i), p_i) + \frac{\lambda}{2} \int_0^1 \left\langle \frac{D^2 \gamma}{dt^2}, \frac{D^2 \gamma}{dt^2} \right\rangle dt,$$

where Γ_2 is the Sobolev space $H^2([0, 1], \mathcal{M})$.

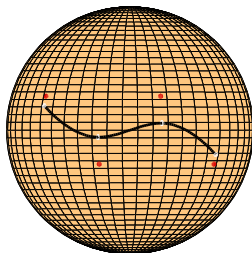
Illustrations on the sphere

$$\text{Objective: } E_2(\gamma) = \frac{1}{2} \sum_{i=0}^N d^2(\gamma(t_i), p_i) + \frac{\lambda}{2} \int_0^1 \left\langle \frac{D^2\gamma}{dt^2}, \frac{D^2\gamma}{dt^2} \right\rangle dt.$$

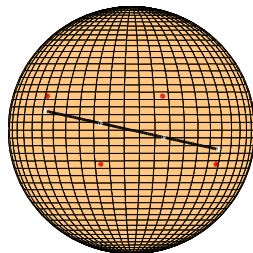
$$\lambda = 10^{-4}$$



$$\lambda = 10^{-3}$$



$$\lambda = 10^0$$



Gradient-descent on set of curves: iterates

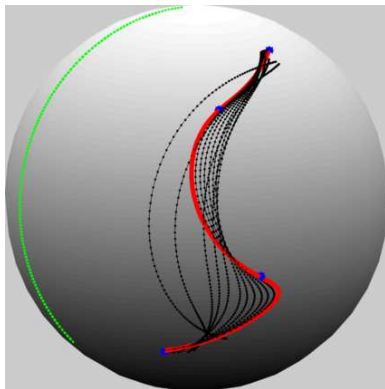
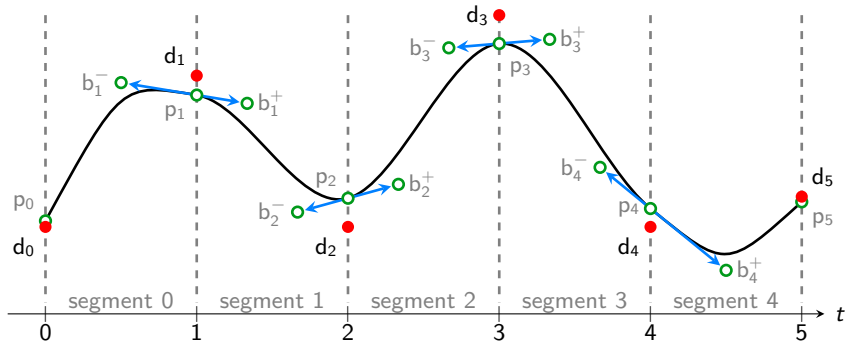


Figure: Iterates γ_k of gradient-descent for E_2 where $\mathcal{M} = \text{sphere}$.

Objective:

$$E_2 : \Gamma_2 \rightarrow \mathbb{R} : E_2(\gamma) = \frac{1}{2} \sum_{i=0}^N d^2(\gamma(t_i), p_i) + \frac{\lambda}{2} \int_0^1 \left\langle \frac{D^2\gamma}{dt^2}, \frac{D^2\gamma}{dt^2} \right\rangle dt.$$

\mathcal{C}^1 -piecewise Bézier smoothing (in \mathbb{R}^n)

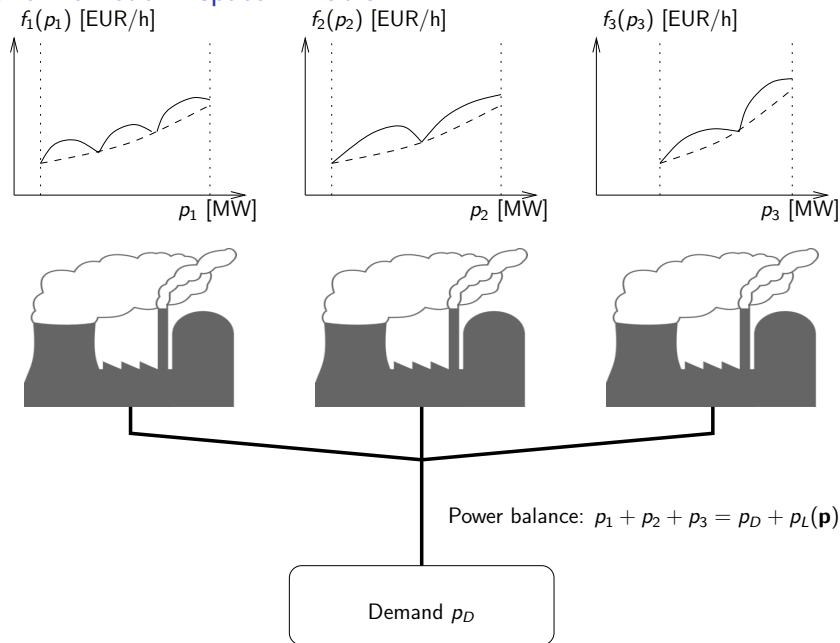
Unknowns: b_i^-, b_i^+, p_i .

C^1 -piecewise Bézier smoothing (in shape space)

VIDEO

Manifolds in power system management

Economic Load Dispatch Problem



Economic Load Dispatch Problem: mathematical model

- ▶ Cost function:

$$\begin{aligned}
 f_{\text{T}}(\mathbf{p}) &= \sum_{i=1}^n f_i(p_i), \quad \mathbf{p} = [p_1, p_2, \dots, p_n]^{\top} \\
 &= \sum_{i=1}^n a_i p_i^2 + b_i p_i + c_i + |d_i \sin [e_i (p_i^{\min} - p_i)]|
 \end{aligned}$$

- ▶ Generator capacity constraints:

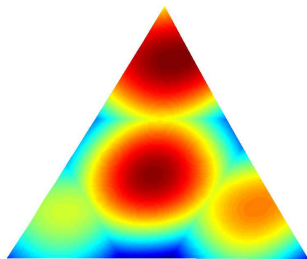
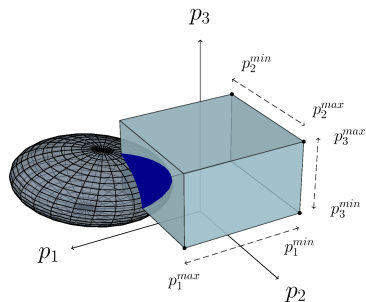
$$p_i^{\min} \leq p_i \leq p_i^{\max}, \quad i = 1, \dots, n$$

- ▶ Real power balance constraint:

$$\begin{aligned}
 \sum_{i=1}^n p_i &= p_{\text{D}} + p_{\text{L}}(\mathbf{p}), \\
 p_{\text{L}}(\mathbf{p}) &= \sum_{i=1}^n \sum_{j=1}^n p_i B_{ij} p_j + \sum_{i=1}^n b_i^0 p_i + b^{00}
 \end{aligned}$$

Our first approach

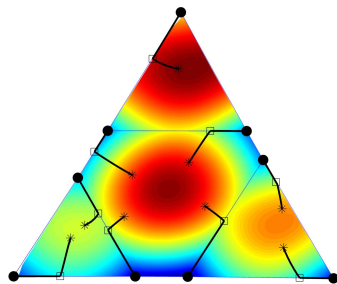
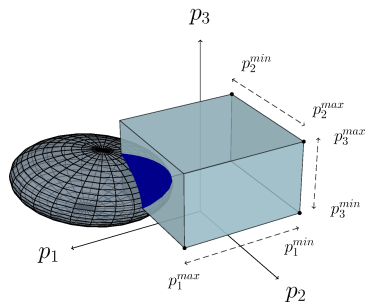
Exploit the geometry of the problem



Reference: Borckmans et al. [BSBA13].

Our first approach

Exploit the geometry of the problem

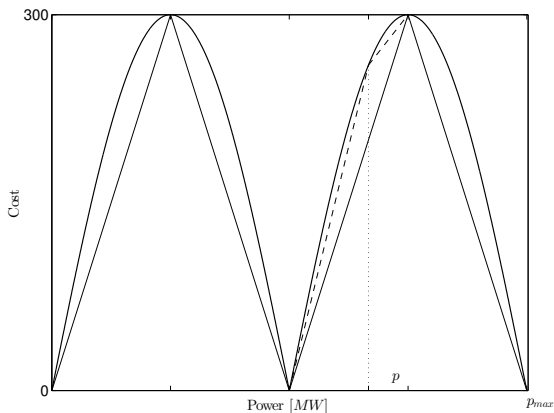


Reference: Borckmans et al. [BSBA13].

New approach

Mixed Integer Programming formulation with guaranteed optimality

- ▶ Using a MIQP solver, compute the global optimum of a *quadratic + piecewise-linear* under-approximation of the true *quadratic + piecewise-concave* cost function.
- ▶ Refine the under-approximation by adding a new break point at the computed global optimum, and repeat.



New approach

Mixed Integer Programming formulation with guaranteed optimality

$$\min_{p,k,t,\chi,\eta} \sum_i \left[a_i p_i^2 + b_i p_i + c_i + \sum_{j=1}^{m_i} (\alpha_{ij} \chi_{ij} + \beta_{ij} \eta_{ij}) \right]$$

$$\text{s.t. } \sum_i p_i = D$$

$$p_i^{\min} \leq p_i \leq p_i^{\max}$$

$$X_{i,j} \eta_{i,j} \leq \chi_{i,j} \leq X_{i,j+1} \eta_{i,j}$$

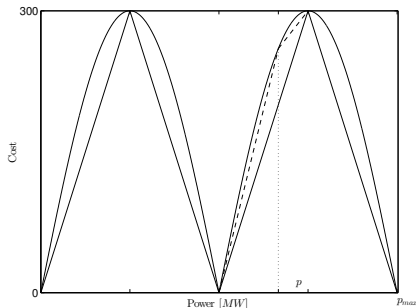
$$p_i = \sum_{j=1}^{m_i} \chi_{i,j}, \quad \sum_{j=1}^{m_i} \eta_{i,j} = 1$$

$$\chi_{i,j} \in \mathbb{R}^+, \quad \eta_{i,j} \in \{0, 1\}$$

$$p_i \in \mathbb{R}$$

New approach

Mixed Integer Programming formulation with guaranteed optimality



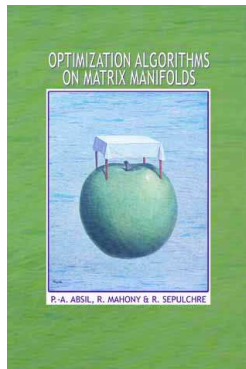
Outcomes:

- ▶ Master's thesis of Michaël Azzam (2013-2014).
- ▶ Master 1 project of Benoît Sluysmans and Nicolas Stevens (2015).
- ▶ Paper showing global convergence for any *quadratic + piecewise-concave* cost function.
- ▶ Code publicly available.

Conclusion: A Three-Step Approach

- ▶ Formulation of the computational problem as a geometric optimization problem.
 - ▶ Smooth constrained set.
 - ▶ Invariances.
- ▶ Generalization of optimization algorithms on abstract manifolds.
- ▶ Exploit flexibility and additional structure to build numerically efficient algorithms.

A 2nd edition?



To be added:

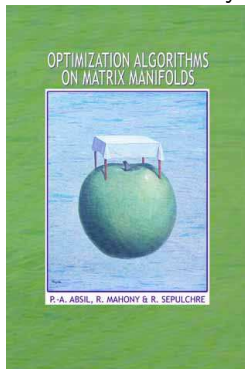
- ▶ Trust-region method without Hessian: Huang et al. [HAG14], Boumal [Bou15].
- ▶ Convergence analysis of Riemannian CG: Sato [Sat16].
- ▶ Constructing retractions: Absil & Malick [AM12].
- ▶ Constructing vector transports: Zhu [Zhu16], Huang et al [HAG17].
- ▶ Low-rank manifold: see work by Bamdev Mishra, Bart Vandereycken...

THE END




Optimization Algorithms on Matrix Manifolds





P.-A. Absil, R. Mahony, R. Sepulchre






Princeton University Press, January 2008












1. Introduction
2. Motivation and applications
3. Matrix manifolds: first-order geometry
4. Line-search algorithms
5. Matrix manifolds: second-order geometry
6. Newton's method
7. Trust-region methods
8. A constellation of superlinear algorithms



-  P.-A. Absil, C. G. Baker, and K. A. Gallivan, *Trust-region methods on Riemannian manifolds*, *Found. Comput. Math.* **7** (2007), no. 3, 303–330.
-  P.-A. Absil and Jérôme Malick, *Projection-like retractions on matrix manifolds*, *SIAM J. Optim.* **22** (2012), no. 1, 135–158.
-  P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization algorithms on matrix manifolds*, Princeton University Press, Princeton, NJ, 2008.
-  Nicolas Boumal, P.-A. Absil, and Coralia Cartis, *Global rates of convergence for nonconvex optimization on manifolds*, *IMA Journal of Numerical Analysis* (2016).
-  Ian Brace and Jonathan H. Manton, *An improved BFGS-on-manifold algorithm for computing weighted low rank approximations*, *Proceedings of the 17h International Symposium on Mathematical Theory of Networks and Systems*, 2006, pp. 1735–1738.

-  Nicolas Boumal, *Riemannian trust regions with finite-difference hessian approximations are globally convergent*, Geometric Science of Information: Second International Conference, GSI 2015, Palaiseau, France, October 28-30, 2015, Proceedings (Frank Nielsen and Frédéric Barbaresco, eds.), Springer International Publishing, Cham, 2015, pp. 467–475.
-  Pierre B. Borckmans, S. Easter Selvan, Nicolas Boumal, and P.-A. Absil, *A Riemannian subgradient algorithm for economic dispatch with valve-point effect*, J. Comput. Applied. Math. **255** (2013), 848–866.
-  D. Gabay, *Minimizing a differentiable function over a differential manifold*, J. Optim. Theory Appl. **37** (1982), no. 2, 177–219. MR MR663521 (84h:49071)
-  B. Gao, X. Liu, X. Chen, and Y. Yuan, *A new first-order algorithmic framework for optimization problems with orthogonality constraints*, SIAM Journal on Optimization **28** (2018), no. 1, 302–332.

-  Gene H. Golub and Qiang Ye, *An inverse free preconditioned Krylov subspace method for symmetric generalized eigenvalue problems*, SIAM J. Sci. Comput. **24** (2002), no. 1, 312–334.
-  Wen Huang, P.-A. Absil, and K. A. Gallivan, *A Riemannian symmetric rank-one trust-region method*, Mathematical Programming, Series A **150** (2014), no. 2, 179–216.
-  Wen Huang, P.-A. Absil, and K. A. Gallivan, *Intrinsic representation of tangent vectors and vector transports on matrix manifolds*, Numerische Mathematik **136** (2017), no. 2, 523–543.
-  Jiang Hu, Xin Liu, Zaiwen Wen, and Yaxiang Yuan, *A brief introduction to manifold optimization*, 2019.
-  Bruno Iannazzo and Margherita Porcelli, *The Riemannian Barzilai-Borwein method with nonmonotone line-search and the karcher mean computation*, December 2015.

-  Bruno Iannazzo and Margherita Porcelli, *The riemannian barzilai-borwein method with nonmonotone line search and the matrix geometric mean computation*, IMA Journal of Numerical Analysis **38** (2018), no. 1, 495–517.
-  David G. Luenberger, *Introduction to linear and nonlinear programming*, Addison-Wesley, Reading, MA, 1973.
-  Hiroyuki Sato, *A Dai-Yuan-type Riemannian conjugate gradient method with the weak Wolfe conditions*, Computational Optimization and Applications **64** (2016), no. 2, 101–118.
-  Michael Shub, *Some remarks on dynamical systems and numerical analysis*, Proc. VII ELAM. (L. Lara-Carrero and J. Lewowicz, eds.), Equinoccio, U. Simón Bolívar, Caracas, 1986, pp. 69–92.
-  B. Savas and L.-H. Lim, *Quasi-Newton methods on Grassmannians and multilinear approximations of tensors*, SIAM J. Sci. Comput. **32** (2010), no. 6, 3352–3393.

-  Steven Thomas Smith, *Geometric optimization methods for adaptive filtering*, Ph.D. thesis, Division of Applied Sciences, Harvard University, Cambridge, MA, May 1993.
-  Steven T. Smith, *Optimization techniques on Riemannian manifolds*, Hamiltonian and gradient flows, algorithms and control (Anthony Bloch, ed.), Fields Inst. Commun., vol. 3, Amer. Math. Soc., Providence, RI, 1994, pp. 113–136. MR MR1297990 (95g:58062)
-  Lorenzo Stella, Andreas Themelis, and Panagiotis Patrinos, *Forward-backward quasi-Newton methods for nonsmooth optimization problems*, 2016.
-  Constantin Udriște, *Convex functions and optimization methods on Riemannian manifolds*, Mathematics and its Applications, vol. 297, Kluwer Academic Publishers Group, Dordrecht, 1994. MR MR1326607 (97a:49038)

-  Xiaojing Zhu, *A Riemannian conjugate gradient method for optimization on the Stiefel manifold*, Computational Optimization and Applications (2016).
-  Hongyi Zhang and Suvrit Sra, *First-order methods for geodesically convex optimization*, 2016.