Optimization and Algorithms on Riemannian Manifolds

Peng Chen

April 21, 2025

Outline

Motivation and Examples

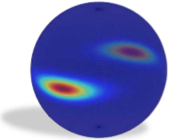
First Order Optimization on Manifolds

Geodesically Convex Optimization

Motivation and Examples

Why Manifolds: Learning Perspective

Surfaces

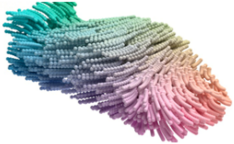


Distributions

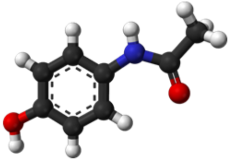
Graphs / Networks

Functions on Manifolds

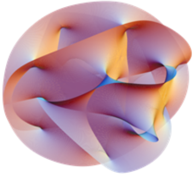
Hyperbolic spaces



Hyper-surfaces



Molecules



General manifolds

“Data has Shape and Shape has Meaning”

Why Manifolds: Optimization Perspective

$$\min_{x} f(x) \text{ s.t. } x \in \mathcal{M} \quad \text{v.s.} \quad \min_{x \in \mathcal{M}} f(x)$$

- Constrained v.s. Unconstrained: accuracy, efficiency.
- Non-convexity v.s. Geodesical Convexity: global minimizer.

Constrained Optimization v.s. Riemannian Optimization

$$\min_x f(x) \text{ s.t. } \underbrace{x \in \mathcal{M}}_{\text{constraint on parameter}}$$

- Linear spaces: unconstrained, linear equality constraints
- Low rank (matrices, tensors): recommender systems, large scale Lyapunov equations
- Orthonormality (Grassmann, Stiefel, rotations): dictionary learning, SfM, SLAM, PCA, ICA, SBM, Electr. Struct. Comp.
- Positivity (positive definiteness, positive orthant): metric learning, Gaussian mixtures, diffusion tensor imaging
- Symmetry (quotient manifolds): invariance under group actions

Constrained Optimization v.s. Riemannian Optimization

$$\min_{x \in \mathcal{M}} f(x) \qquad \text{(unconstrained optimization)}$$

- We can use unconstrained optimization tools (gradient descent, Newton etc.).
- No need to consider Lagrange multipliers or penalty functions.
- Theoretical guarantees usually transfer from Euclidean space to Riemannian manifolds
- Can be cheaper in terms of resource use depending upon the application.

We focus on **embedded submanifolds** of **linear spaces**.

Largest Eigenvalue: Sphere

$$\text{largest eigenvalue: } \max_{x \in \mathbb{S}^{n-1}} f(x) = \langle x, Ax \rangle$$

- $A \in \mathbb{R}^{n \times n}$ with $A^\top = A$ is a symmetric matrix.
- Unit sphere $\mathbb{S}^{n-1} = \{x \in \mathbb{R}^n : x^\top x = 1\}$ is an embedded submanifold of $\mathbb{R}^n$.

Largest Singular Value: Product of Spheres

largest singular value: $$\max_{x \in \mathbb{S}^{m-1}, y \in \mathbb{S}^{n-1}} f(x, y) = \langle x, My \rangle$$

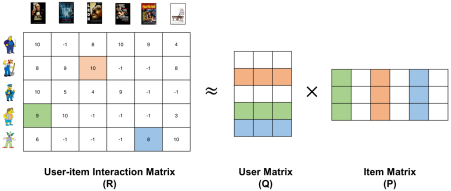
- $M \in \mathbb{R}^{m \times n}$ is a data matrix.
- The Cartesian product $\mathbb{S}^{m-1} \times \mathbb{S}^{n-1}$ is an embedded submanifold of $\mathbb{R}^m \times \mathbb{R}^n$.

Principal Component Analysis: Stiefel

$$\text{top-k eigenspace: } \max_{U \in \mathrm{St}(d,k)} f(U) = \sum_{i=1}^{k} \langle XX^\top u_i, u_i \rangle = \mathrm{tr}\left(U^\top XX^\top U\right).$$

- $X \in \mathbb{R}^{d \times n}$ is the collection of n centered data points in $\mathbb{R}^d$.
- $\mathrm{St}(d,k) = \{U \in \mathbb{R}^{d \times k} : U^\top U = I_k\}$ is the Stiefel manifold embedded in $\mathbb{R}^{d \times k}$.
- The collection of k top eigenvectors of $XX^\top$ yields a global optimum.
- However, the optimization perspective matters when sparsity or robustness are considered additionally.

Low-rank Matrix Completion: Fixed-rank Manifold

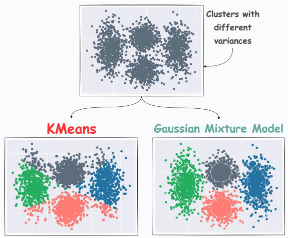


low-rank matrix completion

$$\min_{X \in \mathbb{R}_r^{m \times n}} f(X) = \sum_{(i,j) \in \Omega} (X_{ij} - M_{ij})^2$$

- $M \in \mathbb{R}^{m \times n}$ is a partially (entries in Ω) observed matrix.
- $\mathbb{R}_r^{m \times n} = \{X \in \mathbb{R}^{m \times n} : \text{rank}(X) = r\}$ is the fixed-rank manifold.
- One typical application is personalized recommendation.

Gaussian Mixture Models: Positive Definite Matrices



classic formulation:

$$\max_{\alpha\in\Delta,\{\mu_j,\Sigma_j\succ 0\}_{j=1}^K} \sum_{i=1}^n \log\left[\sum_{j=1}^K \alpha_j p_{\mathcal{N}}(x_i;\mu_j,\Sigma_j)\right]$$

manifold formulation:

$$\max_{\{S_j\succ 0\}_{j=1}^K,\{\eta_j\}_{j=1}^{K-1}} \sum_{i=1}^n \log\left[\sum_{j=1}^K \frac{\exp(\eta_j)}{\sum_{k=1}^K \exp(\eta_k)} q_{\mathcal{N}}(y_i;S_j)\right]$$

- $x_i \in \mathbb{R}^d, i\in[n]$ are the samples, $y_i = (x_i^\top, 1)^\top \in \mathbb{R}^{d+1}$ are augmented samples.
- $p_{\mathcal{N}}(x_i;\mu_j,\Sigma_j)$ is the Gaussian density and $q_{\mathcal{N}}(y_i;S_j) = \sqrt{2\pi e}\cdot p_{\mathcal{N}}(y_i;0,S_j)$
- Parameters lie in the product manifold $\left(\Pi_{j=1}^K \mathbb{P}^d\right)\times\mathbb{R}^{K-1}$.
- Hosseini and Sra 2015 showed the robustness of Riemannian manifold optimization over EM algorithm.

First Order Optimization on Manifolds

Optimization on the Euclidean Space

$$x_{k+1} = \underbrace{x_k}_{\text{current iterate}} \overbrace{+}^{\text{movement}} \Delta x_k = x_k + \tau_k d_k$$

- Steepest (Gradient) descent:

$$x_{k+1} = x_k - \tau_k \nabla f(x_k).$$

- Newton method:

$$x_{k+1} = x_k - \left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k).$$

- Other methods to compute direction d_k and step size τ_k.

A Toy Manifold Optimizer: Projected Gradient Descent

PGD iteration:

1. perform gradient descent:

$$x_{k+\frac{1}{2}} = x_k - \tau_k \nabla f(x_k).$$

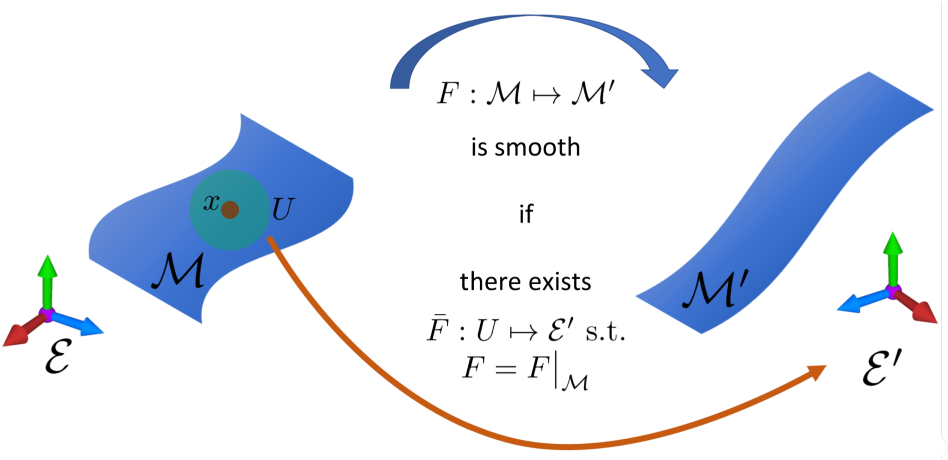
2. project on the manifold:

$$x_{k+1} = \Pi_{\mathcal{M}} \left(x_{k+\frac{1}{2}} \right).$$

Inaccurate & Inefficient.

Smooth Map

Smoothness via the extension.

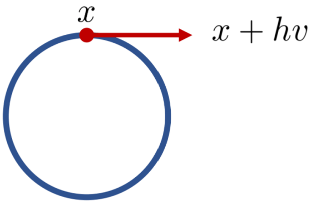


Differential of Smooth Map

- Directional derivative of a smooth function $f : \mathbb{R}^n \to \mathbb{R}$ along a vector v:

$$D_v f = Df(x)[v] := \lim_{h \to 0} \frac{f(x + hv) - f(x)}{h}.$$

- Cannot apply to a smooth manifold $\mathcal{M}$ as $(x + hv)$ might not be belong to $\mathcal{M}$.



Differential of Smooth Map

- For $\bar{F} : \mathcal{E} \to \mathcal{E}'$, it is always defined that

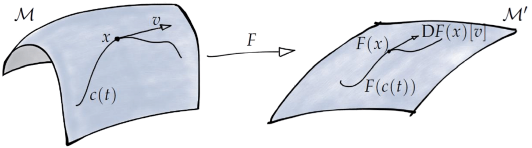
$$D_v \bar{F} = D\bar{F}(x)[v] := \lim_{h \to 0} \frac{\bar{F}(x + hv) - \bar{F}(x)}{h}.$$

- For $F : \mathcal{M} \to \mathcal{M}'$ and $v \in T_x\mathcal{M}$, we write

$$D_v F = D_v \bar{F}.$$

Independent of the choice of the smooth extension!

Differential of Smooth Map

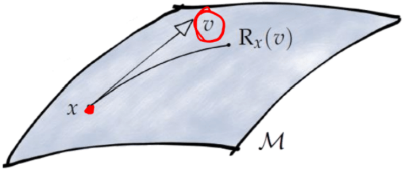


- Equivalently, consider a smooth curve $c(t) \in \mathcal{M}$ with $F(c(t)) \in \mathcal{M}'$ being a curve in $\mathcal{M}'$ passing through $F(x)$ with velocity $DF(x)[v]$.
- Computation via the curve:

$$DF(x)[v] : T_x\mathcal{M} \to T_{F(x)}\mathcal{M}', v \mapsto \frac{d}{dt}F(c(t))|_{t=0} = (F \circ c)'(0).$$

Independent of the choice of the curve!

How to Choose these Curves: Retractions



- Smooth choice of curves over the tangent bundle.
- Maps tangent vectors back to the manifold.
- Defines curves in a given direction.
- A Retraction map $R : T_x\mathcal{M} \to \mathcal{M}$ satisfies:
  1. R is continuously differentiable.
  2. $R_x(0) = x$ (centering).
  3. $DR_x(0)[v] = v$ (local rigidity).
- Choose the curve as $c(t) = T(x, tv) = R_x(tv)$ with $c(0) = x$ and $c'(0) = v$.

Projection as a Retraction

- A retraction on the sphere $\mathbb{S}^{d-1}$:

$$R_x(v) = \frac{x+v}{\|x+v\|}$$

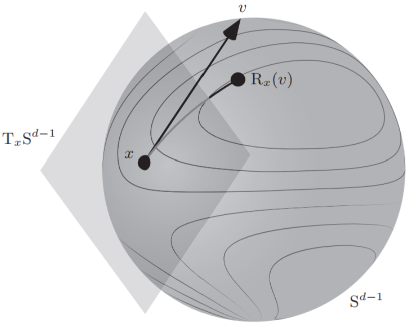


Figure 1: Retraction on the sphere.

Ingredients of Optimization

- Representations for points $x \in \mathcal{M}$, tangent spaces $T_x\mathcal{M}$ and Riemannian metric $g_x(\cdot, \cdot)$.
- A map from the tangent space to the manifold: $R_x : T_x\mathcal{M} \to \mathcal{M}$
- Expressions for $f(x), \mathrm{grad} f(x)$ and $\mathrm{Hess} f(x)$.
- Notion of vector transport for second order methods. (not covered here)

Riemannian Gradient

- Riemannian gradient of $f(x)$ at x is the unique tangent vector in $T_x\mathcal{M}$ satisfying:
$$Df(x)[v] = \langle \mathrm{grad} f(x), v \rangle_x.$$
- If x is a local optimum of f, then $\mathrm{grad} f(x) = 0$.

Riemannian Gradient

- Since $D(f \circ R_x)(0)[v] = Df\left(R_x(0)\right)\left[DR_x(0)[v]\right] = Df(x)[v]$, it holds that

$$\operatorname{grad} f(x) = \operatorname{grad}(f \circ R_x)(0), \forall x \in \mathcal{M}.$$

- Indeed, $f \circ R_x : T_x\mathcal{M} \to \mathbb{R}$ is defined on a Euclidean space (the linear space $T_x\mathcal{M}$ with inner product $\langle \cdot, \cdot \rangle_x$) and thus $\operatorname{grad}(f \circ R_x)$ is the classic gradient.
- $Df(x)[v] = D\bar{f}(x)[v] = \langle v, \operatorname{grad} \bar{f}(x) \rangle$.

Riemannian Gradient

- Observe $T_x\mathcal{M}$ is a subspace of $\mathcal{E}$ and $\operatorname{grad}\bar{f}(x) \in \mathcal{E}$, it can be decompose as

$$\operatorname{grad}\bar{f}(x) = \underbrace{\operatorname{grad}\bar{f}(x)_{\parallel}}_{\text{tangential}} + \underbrace{\operatorname{grad}\bar{f}(x)_{\perp}}_{\text{orthongal}}.$$

- Since $v \in T_x\mathcal{M}$, $\langle v, \operatorname{grad}\bar{f}(x)_{\perp}\rangle = 0$ and thus

$$\begin{aligned}\langle v, \operatorname{grad}f(x)\rangle_x &= \langle v, \operatorname{grad}\bar{f}(x)\rangle_x \\ &= \langle v, \operatorname{grad}\bar{f}(x)_{\parallel} + \operatorname{grad}\bar{f}(x)_{\perp}\rangle \\ &= \langle v, \operatorname{grad}\bar{f}(x)_{\parallel}\rangle\end{aligned}$$

Riemannian Gradient

$$\operatorname{grad} f(x) = \operatorname{grad} \bar{f}(x)_{\|}$$

Steps to compute the Riemannian gradient:

- obtain an expression for the classical gradient: $\operatorname{grad} \bar{f}(x)$
- orthogonally project to the tangent space: $\mathrm{Proj}_x \left(\operatorname{grad} \bar{f}(x) \right)$
  - $\mathrm{Proj}_x(\cdot)$is the orthogonal projection $\Pi_{T_x \mathcal{M}}(\cdot)$.

Example: Rayleigh Quotient on the Sphere

$$f : \mathbb{S}^{d-1} \to \mathbb{R}, x \mapsto x^\top A x.$$

- Extension: $\bar{f} : \mathbb{R}^d \to \mathbb{R}, x \mapsto x^\top A x$ with $D\bar{f}(x)[v] = \langle 2Ax, v\rangle, \operatorname{grad} \bar{f}(x) = 2Ax$.
- Tangent space: $T_x\mathbb{S}^{d-1} = \left\{v \in \mathbb{R}^d : \langle x, v\rangle = 0\right\}$.
- Projection: $\operatorname{Proj}_x(u) = (I - xx^\top)(u)$.
- Riemannian gradient: $\operatorname{grad} f(x) = \operatorname{Proj}_x(\operatorname{grad} \bar{f}(x)) = 2\left[Ax - (x^\top Ax)x\right]$

$$\operatorname{grad} f(x) = 0 \iff Ax = \underbrace{(x^\top Ax)}_{\text{scalar}}\, x \qquad \text{(eigen vector)}$$

Optimal Points

Given a cost function $f : \mathcal{M} \to \mathbb{R}$ on a manifold, we aim to solve:

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

Def.: $x \in \mathcal{M}$ is a global minimum if $f(x) \leq f(y)$ for all $y \in \mathcal{M}$.

Def.: $x \in \mathcal{M}$ is a local minimum if there exists a neighbourhood $x \in U \subset \mathcal{M}$ such that $f(x) \leq f(y)$ for all $y \in U$.

Def.: $x \in \mathcal{M}$ is critical or stationary for $f : \mathcal{M} \to \mathbb{R}$ if $(f \circ c)'(0) \geq 0$ for all smooth curves c on $\mathcal{M}$ such that $c(0) = x$.

First Order Optimality Condition

Theorem 1

1) *If x is a local minimum, then it is critical.*
2) *On a Riemannian manifold, x is critical iff* $\mathrm{grad} f(x) = 0$.

Sketched proof for 2):
Identity: $(f \circ c)'(0) = Df(x)[v] = \langle \mathrm{grad} f(x), v \rangle_x$ for $c : c(0) = x, c'(0) = v$.
If $\mathrm{grad} f(x) = 0$, then $(f \circ c)'(0) = 0 \geq 0, \forall c$.
If $(f \circ c)'(0) \geq 0$, then

$$\langle \mathrm{grad} f(x), v \rangle_x \geq 0, \forall v \in \underbrace{T_x\mathcal{M}}_{\text{linear space}} \implies \mathrm{grad} f(x) = 0.$$

Riemannian Gradient Descent

$$\text{RGD: } x_{k+1} = R_{x_k}(-\tau_k \operatorname{grad} f(x_k))$$

Taylor perspective: the composition $f \circ R_x : T_x\mathcal{M} \to \mathbb{R}$ is defined on a linear space and has a **Taylor expansion**:

$$\begin{aligned} f(R_x(v)) &= f(R_x(0)) + \langle \operatorname{grad}(f \circ R_x)(0), v \rangle + O(\|v\|^2) \\ &= f(x) + \langle \operatorname{grad} f(x), v \rangle_x + O(\|v\|_x^2). \end{aligned}$$

Convergence Theory

Proposition 1

*Let f be a smooth and **lower bounded** (by f_{low}) function on a Riemannian manifold $\mathcal{M}$. Let $x_0, x_1, x_2, \cdots$ be iterates satisfying **sufficient decrease**:*

$$f(x_k) - f(x_{k+1}) \geq c\|\mathrm{grad} f(x_k)\|^2$$

with constant c. Then,

$$\lim_{k\to\infty} \|\mathrm{grad} f(x_k)\| = 0.$$

In particular, all accumulation points (if any) are critical points. Furthermore, for all $k \geq 1$, there exists k in $0, \cdots, K-1$ such that

$$\|\mathrm{grad} f(x_k)\| \leq \sqrt{\frac{f(x_0) - f_{\text{low}}}{c}} \frac{1}{\sqrt{K}}.$$

Convergence Theory

Sketched proof of Proposition 1:

- The $\mathcal{O}\left(1/\sqrt{K}\right)$ follows from a standard telescoping sum argument:

$$f(x_0) - f_{\text{low}} \geq f(x_0) - f(x_K) \geq \sum_{k=0}^{K-1} f(x_k) - f(x_{k+1}) \geq Kc \min_{0\leq k\leq K-1} \|\operatorname{grad} f(x_k)\|^2.$$

- The limit statement can be derived as follows

$$f(x_0) - f_{\text{low}} \geq \sum_{k=0}^{\infty} \underbrace{f(x_k) - f(x_{k+1})}_{\text{nonnegative}} \Longrightarrow 0 = \lim_{k\to\infty} f(x_k) - f(x_{k+1}) \geq c\|\operatorname{grad} f(x_k)\|^2.$$

- Continuity gives the stationarity of accumulation points ($x_{(k)} \to x^*$):

$$0 = \lim_{k\to\infty} \|\operatorname{grad} f(x_k)\| = \lim_{k\to\infty} \|\operatorname{grad} f(x_{(k)})\| = \|\operatorname{grad} f(x^*)\|.$$

Guarantee Sufficient Decrease

regular condition: $f\left(R_x(s)\right) \leq f(x) + \langle \operatorname{grad} f(x), s \rangle + \frac{L}{2}\|s\|^2, \forall (x, s) \in S \subset T\mathcal{M}.$

- Under the above regular condition, if $\tau_k \in [\tau_{\min}, \tau_{\max}] \subset (0, 2/L)$, the sufficient decrease in Proposition 1 is obtained with

$$c = \min\left\{\tau_{\min} - \frac{L}{2}\tau_{\min}^2, \tau_{\max} - \frac{L}{2}\tau_{\max}^2\right\}.$$

- In particular, for constant $\tau_k = 1/L$ we have $c = 1/(2L)$.
- The regular condition is similar to the Lipschitz smoothness condition in Euclidean optimization.

Backtracking Line Search

- In practice, an appropriate constant L is seldom known.
- A blindly large L forcing small steps is evidently not necessary.
- Only the **local behavior** of f around x_k matters to ensure sufficient decrease.
- A common adaptive strategy to pick τ_k for RGD is backtracking line search.

Backtracking Line Search

Algorithm 1 Backtracking line search

Input: $x \in \mathcal{M}, \bar{\tau} > 0, \alpha, \beta \in (0, 1)$.
Set $\tau \leftarrow \bar{\tau}$
while $f(x) - f(R_x(-\tau \mathrm{grad} f(x))) < \beta\tau\|\mathrm{grad} f(x)\|^2$ **do**
  Set $\tau \leftarrow \alpha\tau$
end while
Output: τ.

- Backtracking line search starts with an intial $\bar{\tau}$ and iteratively reduces it by a factor α until the **Armijo–Goldstein** condition is satisfied such that

$$f(x) - f(R_x(-\tau \mathrm{grad} f(x))) \geq \beta\tau\|\mathrm{grad} f(x)\|^2.$$

- In practice, we can take $\alpha = \frac{1}{2}, \beta = 10^{-4}$.
- Under the regularity condition, backtracking line search guarantees sufficient decrease, with a constant c which depends on various factors.

A Generic Riemannian Optimization Algorithm

Algorithm 2 Generic Riemannian Optimization Algorithm

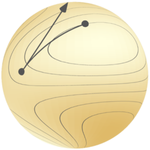
Input: A Riemannian manifold $\mathcal{M}$, a retraction operator R.
while x_k does not suficiently minimize f **do**
  Pick a gradient related descent direction $\eta_k \in T_{x_k}\mathcal{M}$.
  Choose a retraction $R_{x_k} : T_{x_k}\mathcal{M} \to \mathcal{M}$.
  Choose a step length $\tau_k \in \mathbb{R}$.
  Set $x_{k+1} \leftarrow R_{x_k}(\tau_k \eta_k)$.
  $k \leftarrow k + 1$.
end while
Output: x_k.

Library: Manopt

Manopt Home Tutorial Learning Downloads Forum About

Manopt

Toolboxes for optimization on manifolds and linear spaces



Optimization on manifolds is a versatile framework for continuous optimization.

It encompasses optimization over vectors and matrices,

and adds the possibility to optimize over curved spaces to handle constraints and symmetries such as orthonormality, low rank, positivity and invariance under group actions.

Manopt makes it easy.

Download GitHub Mailing list

Matlab

This website is the home of the Matlab version of Manopt. Install from downloads or from our GitHub repository, then run a first example.

Python

The PyManopt website houses the Python version of Manopt and its documentation. Also check out the GitHub repository.

Julia

The Manopt.jl website hosts the Julia version of Manopt and its documentation. The GitHub repository has both Manopt.jl and Manifolds.jl.

PyManopt: Code Example

```python
import autograd.numpy as anp
import pymanopt

dim = 3
manifold = pymanopt.manifolds.Sphere(dim)  # specify the manifold

matrix = ...  # data matrix
@pymanopt.function.autograd(manifold)  # Riemannian autograd related to manifold
def cost(point):
    return - point @ matrix @ point  # Rayleigh quotient for largest eigenvector

problem = pymanopt.Problem(manifold, cost)
optimizer = pymanopt.optimizers.SteepestDescent()  # solve with RGD algorithm
result = optimizer.run(problem)
```

Geodesically Convex Optimization

Why Convexity

In a linear space $\mathcal{E}$, a minimization problem

$$\min_{x \in S} f(x)$$

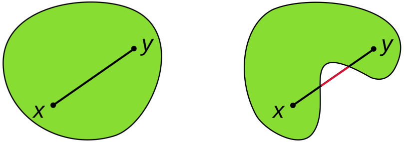
is convex if the search space S and the cost function f are convex.

Convex optimization has advantages:

1. Local minima are global minima.
2. This comes up in applications and it's easy to spot.
3. There exist good algorithms.

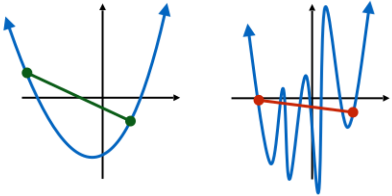
Convexity in Linear Space $\mathbb{R}^n$

- A set $S \subset \mathbb{R}^n$ is convex if $x, y \in S \implies (1-t)x + ty \in S, \forall t \in [0,1]$.



- A function $f : S \to \mathbb{R}$ is convex if its domain S is convex and

$$f((1-t)x + ty) \leq (1-t)f(x) + tf(y), \forall x, y \in S, t \in [0,1]$$



convex nonconvex

Convexity on Riemannian Manifold $\mathcal{M}$

extend the convexity to $\mathcal{M}$ while preserving "local min$\Longrightarrow$ global min"

- A set $S \subset \mathcal{M}$ is g-convex if $\forall x, y \in \mathcal{M}$, there exists a geodesic segment $c : [0, 1] \to \mathcal{M}$ such that $c(0) = x, c(1) = y$ and $c(t) \in S, \forall t \in [0, 1]$.
- A function $f : S \to \mathbb{R}$ is g-convex if S is g-convex and $f \circ c : [0, 1] \to \mathbb{R}$ is convex, i.e., for each geodesic segment c in $\mathcal{M}$ with $c(0) = x, c(1) = y$, it holds that
$$f(c(t)) \leq (1 - t)f(x) + tf(y), \forall t \in [0, 1].$$
- Strict (strong) g-convexity of f can be defined similarly via the strict (strong) convexity of $f \circ c$.

Convexity on Riemannian Manifold $\mathcal{M}$

g-convex sets:

- Ex. 1: If $\mathcal{M}$ is complete and connected, then $S := \mathcal{M}$ is g-convex.
- Ex. 2: For any $y \in \mathbb{S}^{d-1}$, the set $S := \{x \in \mathbb{S}^{d-1} : \mathrm{dist}(x, y) \leq r\}$ is g-convex.

g-convex functions:

- Ex. 1: $f(x) = \frac{1}{2}\mathrm{dist}(x, y)^2$ is g-convex on the domain $\{x \in \mathcal{M} : \mathrm{dist}(x, y) \leq r\}$ provided r is small enough.

Properties:

1. Sublevel sets of g-convex functions are g-convex sets.
2. Intersections of such sublevel sets are g-convex sets.
3. Sums of nonnegatively scaled g-convex functions are g-convex.
4. The pointwise maximum of g-convex functions is g-convex.

Geodesically Convex Optimization

$$\min_{x \in S} f(x)$$

- It is a geodesically convex optimization problem if both S and f are g-convex.
- **Fact:** If x is a local minimum, then it is a global minimum.

Sketched proof:
Suppose in contradiction that there exists $y \in S$ such that $f(y) < f(x)$.
S is g-convex:

$$\exists c : [0,1] \to \mathcal{M} \text{ s.t. } c(0) = x, c(1) = y, c(t) \in S, \forall t \in [0,1]$$

f is g-convex:

$$f(c(t)) \leq (1-t)f(x) + tf(y) < f(x).$$

Taking $t \to 0$ contradicts to the local optimality of x.

Polyak-Łojasiewicz Condition

Definition 2

Let $f : \mathcal{M} \to \mathbb{R}$ be differentiable on a Riemannian manifold $\mathcal{M}$. We say f satisfies the Polyak–Łojasiewicz condition with constant $\mu > 0$ on a set $S \subset \mathcal{M}$ if

$$f(x) - f^* \leq \frac{1}{2\mu}\|\mathrm{grad} f(x)\|_x^2 \text{ for all } x \in S$$

where $f^* := \inf_{x \in S} f(x)$.

- PŁ holds for geodesically strongly convex f.
- **Intuition:** Within S, the squared gradient norm bounds the optimality gap.

Linear Convergence

Theorem 3

Let $f : \mathcal{M} \to \mathbb{R}$ be differentiable on a Riemannian manifold $\mathcal{M}$. Consider a sequence of points $x_0, x_1, \cdots$ on $\mathcal{M}$. Assume the following hold for all k:

1. *Sufficient decrease:* $f(x_k) - f(x_{k+1}) \geq \frac{1}{2L} \|\mathrm{grad} f(x_k)\|_{x_k}^2$
2. *PŁ:* $f(x_k) - f^* \leq \frac{1}{2\mu} \|\mathrm{grad} f(x_k)\|_{x_k}^2$

Then, $f(x_k) - f^* \leq (1 - \frac{\mu}{L})^k (f(x_0) - f^*), \forall k.$

Linear Convergence

Sketched proof:

$$
\begin{aligned}
f(x_{k+1}) - f^* &= f(x_{k+1}) - f(x_k) + f(x_k) - f^* \\
&\leq -\frac{1}{2L} \|\mathrm{grad} f(x_k)\|_{x_k}^2 + f(x_k) - f^* && \text{(sufficient decrease)} \\
&\leq -\frac{2\mu}{2L} \left[f(x_k) - f^*\right] + f(x_k) - f^* && \text{(ŁP)} \\
&= \left(1 - \frac{\mu}{L}\right) \left[f(x_k) - f^*\right].
\end{aligned}
$$

References I

- Absil, P-A, Robert Mahony, and Rodolphe Sepulchre (2009). "Optimization algorithms on matrix manifolds". In: *Optimization Algorithms on Matrix Manifolds*. Princeton University Press.
- Boumal, Nicolas (2023). *An introduction to optimization on smooth manifolds.* Cambridge University Press.
- Hosseini, Reshad and Suvrit Sra (2015). "Matrix manifold optimization for Gaussian mixtures". In: *Advances in neural information processing systems* 28.
- Rebjock, Quentin and Nicolas Boumal (2023). "Fast convergence to non-isolated minima: four equivalent conditions for C^2 functions". In: *arXiv preprint arXiv:2303.00096*.
- Zhang, Hongyi and Suvrit Sra (2016). "First-order methods for geodesically convex optimization". In: *Conference on learning theory*. PMLR, pp. 1617–1638.