Geometric Optimization Algorithms for Matrix Completion

Li Zhizhong, Depart. of Math., Peking University

2014/9/10

A Quick Example:
An Ant Living on a Globe

- If the ant want to optimize a function on this globe, say, find the most attractive point, it may use the strategy of gradient ascend provided that the attraction function is differential.

- This talk aims at explaining this idea with mathematical rigorous.

*Picture from web

Outline

- In a nutshell, we are going to COPY the various optimization methods in *Numerical Optimization* and PASTE them to the manifold setting, and then apply this idea to solve the *Matrix Completion* problem.
- We will go through all the necessary details of mathematics, so don't worry if you are not familiar with some concepts.

- We are going to talk about:
  - What is a manifold,
  - Review the traditional optimization methods,
  - Show how to do the COPY/PASTE,
  - And application to Matrix Completion.

References

- This talk is based on the following materials:
  - Mishra, B., and R. Sepulchre. "R3MC: A Riemannian Three-Factor Algorithm for Low-Rank Matrix Completion." (2013).
  - Boumal, Nicolas, and Pierre-antoine Absil. "RTRMC: A Riemannian trust-region method for low-rank matrix completion." *Advances in neural information processing systems*. 2011.
  - Mishra, Bamdev, K. Adithya Apuroop, and Rodolphe Sepulchre. "A Riemannian geometry for low-rank matrix completion." *arXiv preprint arXiv:1211.1550* (2012).
  - Vandereycken, Bart. "Low-rank matrix completion by Riemannian optimization---extended version." *arXiv preprint arXiv:1209.3834* (2012).
  - Wen, Zaiwen, Wotao Yin, and Yin Zhang. "Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm." *Mathematical Programming Computation* 4.4 (2012): 333-361.

References

- Warner, Frank W. *Foundations of Differentiable Manifolds and Lie Groups*. 94 Vol. New York: Springer, 1983.
- Hirsch, Morris W. *Differential Topology*. 33 Vol. New York: Springer-Verlag, 1976.
- Petersen, Peter. *Riemannian Geometry*. 171 Vol. New York: Springer, 2006.
- R. Mahony, R. Sepulchre, and P.-A. Absil. *Optimization Algorithms on Matrix Manifolds*. Princeton: Princeton University Press, 2009.
- Edelman, Alan, Tomás A. Arias, and Steven T. Smith. "The geometry of algorithms with orthogonality constraints." *SIAM journal on Matrix Analysis and Applications* 20.2 (1998): 303-353.
- Nocedal, Jorge, and Stephen J. Wright. "Numerical Optimization 2nd." (2006).
- Golub, Gene H., and Charles F. Van Loan. *Matrix computations*. Vol. 3. JHU Press, 2012.

Introduction to Matrix Completion

- Collaborative Filtering
- Matrix Completion

Collaborative Filtering

- *Collaborative filtering* (CF) is a technique used by some recommender systems.
- It is a recommender systems only based on logs of user usages on items.

Picture form wiki

Matrix Completion

- One way to model is using *Matrix Completion* (MC).
- The task of **Matrix Completion** is to recover a *low-rank* matrix which a few entries are observed, possibly with noise.
- Use the assumption that the matrix is low-rank which is based on the premise that only a small number of factors have strong influence.

$$\begin{pmatrix} ? & 1 & ? \\ 1 & 0 & 1 \\ ? & 1 & ? \end{pmatrix}$$

Formulation of Matrix Completion

- We address the problem when the rank is a priori known.
- Given a matrix $X^* \in \mathbb{R}^{n\times m}$ whose entries are given for indices $(i,j) \in \Omega$ where Ω is a subset of all possible indices $\{(i,j): i \in \{1,\dots,n\}, j \in \{1,\dots,n\}\}$ and we find an X who minimize

$$\arg\min_{X\in\mathbb{R}_r^{n\times m}} \frac{1}{2}\|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(X^*)\|_F^2$$

where $\mathbb{R}_r^{n\times m}$ is the set of rank-r $n \times m$ matrices, the function

$$\mathcal{P}_\Omega(X)_{i,j} = \begin{cases} X_{i,j} & \text{if } (i,j) \in \Omega \\ 0 & \text{otherwise} \end{cases}$$

Notes on Matrix Completion

- In general, one cannot expect to be able to recover a low-rank matrix from a sample of its entries.
- Consider the rank-1 matrix M with only one none zero element at the top-right corner:

$$M = \begin{bmatrix} 0 & 0 & \cdots & 1 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}$$

- Under suitable conditions, reconstruction is workable.

Matrix Completion and Geometric Optimization Methods

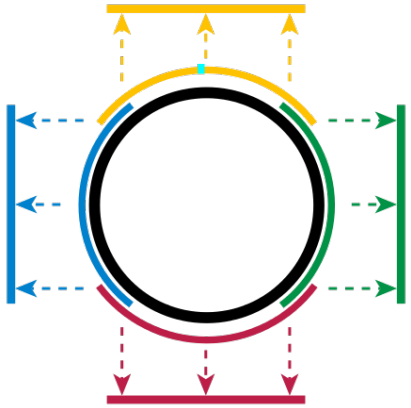
- We have noticed that the search space of the MC problem is the set of fixed-rank matrices $\mathbb{R}_r^{n\times m}$.
- It must be more efficient if we can optimize right on the space of $\mathbb{R}_r^{n\times m}$ rather than the whole matrix space $\mathbb{R}^{n\times m}$.
- Later we will see that $\mathbb{R}_r^{n\times m}$ is a manifold(curved surface) of dimension $(m+n-r)r$ which is much lower than $\dim \mathbb{R}^{n\times m} = mn$ if $r \ll m, n$.
- In fact, geometric optimization methods performs well on MC problem.

Introduction to Riemannian Manifolds

- Differential Manifolds and Riemannian Manifolds
- Product manifolds and Quotient manifolds
- Tangent Space

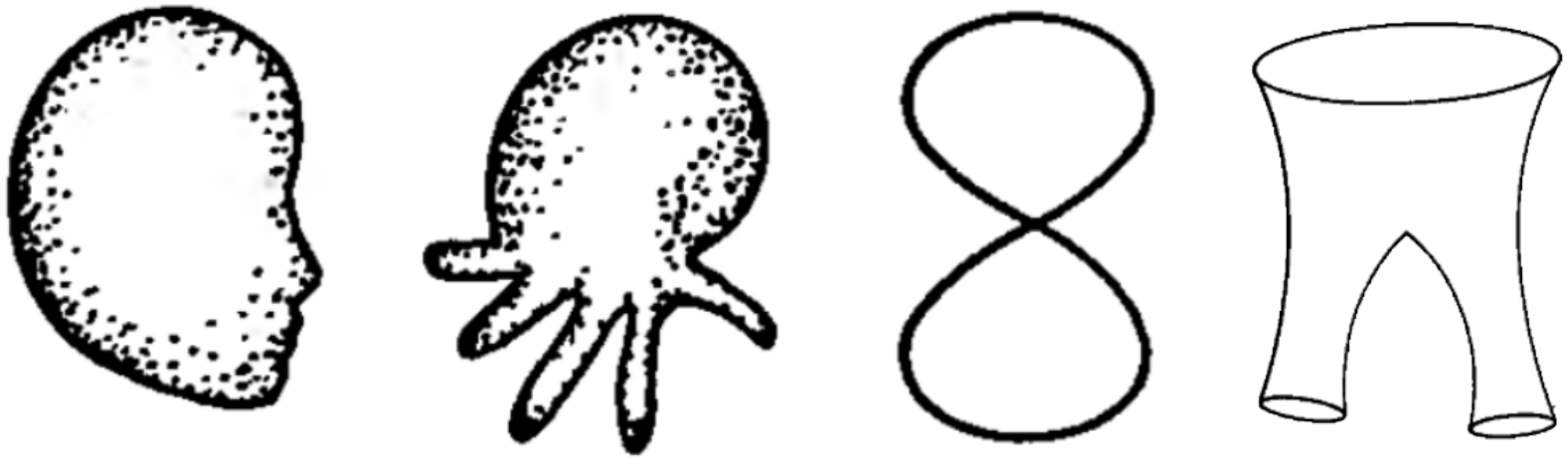
Differential Manifolds

- A **manifold** M is a locally Euclidean topological space. For any point $x \in M$, one can find a neighborhood U homeomorphic to an open subset of $\mathbb{R}^n$.
- This homeomorphism $\phi: U \to \mathbb{R}^n$ provides a **local coordinate**. It gives every point in U a Euclidian coordinate.
- (U, ϕ) is called a **chart**. All the chart satisfy certain compatibility conditions.



Picture form wiki

Differential Manifolds



✓ ✓ × ✓

Note on Manifolds

- By embedding theorems of Whitney and Nash, any smooth manifold(or Riemannian manifold) can be smoothly(or isometrically) embedded into some Euclidean space.
- So, without loss of generality, a **differential manifold** can be seen as a multi-dimensional curved surface in an Euclidean space.
- We use surface and manifold synonymously in this talk.
- The usual Euclidean space is a manifold.

Construct Manifolds 1: Submanifolds

- A subset of a manifold may be a manifold, e.g. surfaces in a space can be seen as a submanifold of that space. They are called **submanifolds**.
- Submanifolds are usually defined by the solution of a set of equations.
- Not arbitrary equations could define a manifold, but there are plenty of successful examples.

Example 1 of Manifold: Spheres

- **n-Sphere** S^n in $n+1$ dimensional Euclidean space is

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

- It is a submanifold of $\mathbb{R}^{n+1}$ defined by the equation $x^T x - 1 = 0$.
- $n = 0$: Two points.
- $n = 1$: Circle in a plane.
- $n = 2$: A normal sphere in our living 3-dim'l world.

Example 2 of Manifold: Matrix Manifolds

- $\mathbb{R}^{m\times n}$: size $m \times n$ matrices, dimension mn.
- $\mathbb{R}_*^{n\times p}$: size $n \times p$ full column rank matrices, dimension np.
- $GL(n)$: size $n \times n$ invertible matrices, dimension n^2.
- $\mathcal{O}(n) = \{X \in \mathbb{R}^{n\times n}: X^T X = I_n\}$: orthogonal groups. $\dim \mathcal{O}(n) = n(n-1)/2$.
- $SO(n) = \{X \in \mathcal{O}(n): \det X = 1\}$: special orthogonal groups. Dim $n(n-1)/2$.

- $\mathbb{R}_*^{n\times 1} = \mathbb{R}_*^n$ is vector space $\mathbb{R}^n$ without origin, or the punctured vector space.
- $\mathbb{R}_*^{n\times n} = GL(n)$

Example 3 of Manifold: Stiefel Manifold

- Stiefel manifold $St(r,n)$ consists of size $n \times r$ matrices with orthonormal columns, i.e.

$$St(r,n) = \{X \in \mathbb{R}^{n\times r} : X^T X = I_r\}$$

- The dimension of $St(r,n)$ is $np - p(p+1)/2$.

- $St(1,n) = S^{n-1}$,
- $St(n,n) = O(n)$,

Fixed Rank Matrices as Manifold

- We use $\mathbb{R}_r^{n\times m}$ to denote rank-r $n \times m$ matrices.
- $\mathbb{R}_r^{n\times m}$ is a submanifold of $\mathbb{R}^{n\times m}$.
- To see this, consider rank-r matrix $A = \begin{pmatrix} B & C \\ D & E \end{pmatrix}$ where B is an $r \times r$ invertible matrix.
- Transform A to $\begin{pmatrix} B & 0 \\ D & -DB^{-1}C + E \end{pmatrix}$ by multiplying $\begin{pmatrix} I & -B^{-1}C \\ 0 & I \end{pmatrix}$

  Then $-DB^{-1}C + E$ must be 0 due to rank constraint.
- So, rank-r matrices near A is the solution of $(m-r)(n-r)$ equations

$$-DB^{-1}C + E = 0$$

- $\dim \mathbb{R}_r^{n\times m} = mn - (m-r)(n-r) = (m+n-r)r.$

Objects Defined on Manifolds

- Smooth functions
- Smooth maps between manifolds, like embedding
- Tangent vectors

Tangent Vector, Tangent Space and Vector fields

- If we see manifolds as submanifold of $\mathbb{R}^n$, then the notion of tangent vector is the tangent vector in $\mathbb{R}^n$.
- A **tangent vector** at x can be represented by the class of smooth curves passing x which share the same tangent vector. This point of view is useful both conceptually and computationally.
- All the tangent vectors at a point $x \in M$ form a vector space, denoted by $T_x M$ and called the **tangent space** of manifold M at point x.
- The dimension of tangent space equals the dimension of the manifold.
- If we specify a tangent vector ξ_x to each point $x \in M$, then we get a **vector field** $\xi \in \mathfrak{X}(M)$, where $\mathfrak{X}(M)$ denote the set of vector fields on M.

Example 1: Tangent Space of a Sphere

- Let $t \mapsto x(t)$ be a curve in the sphere S^{n-1} s.t. $x(0) = x_0$ and $\dot{x}(0) = y$. We want to find the constraint that y satisfies.
- Differentiating the equation

$$x^T(t)x(t) = 1$$

we get

$$\dot{x}^T(t)x(t) + x^T(t)\dot{x}(t) = 0$$

so, y satisfies

$$y^T x_0 = 0$$

- The set $\{y \in \mathbb{R}^n : y^T x_0 = 0\}$ consists the tangent space $T_{x_0}S^{n-1}$.

Example 2: Tangent Space of $St(p,n)$

- Similarly,

$$T_{X_0}St(p,n) = \{Y \in \mathbb{R}^{n\times p} : X_0^T Y + Y^T X_0 = 0\}$$

- Here's another useful characterization of the tangent space of $St(p,n)$.
- Write $\dot{X}(t)$ in the form

$$\dot{X}(t) = X(t)\Omega(t) + X_\perp(t)K(t)$$

where $X_\perp$ is any $n \times (n-p)$ matrix whose column space is the orthogonal complement of the column space of X.

- We'll get the constraint $\Omega(t)^T + \Omega(t) = 0$, so

$$T_{X_0}St(p,n) = \{X\Omega + X_\perp K : \Omega^T = -\Omega, K \in \mathbb{R}^{(n-p)\times p}\}$$

Differential of a Smooth Map

- A smooth map $\phi: M \to N$ can map a curve $t \mapsto x(t)$ in M to a curve $t \mapsto \phi\big(x(t)\big)$ in N.
- So, it can map the tangent vector $\xi \in T_x M$ to the tangent vector $\eta \in T_{\phi(x)} N$.
- This defines the **differential of map** ϕ at point x

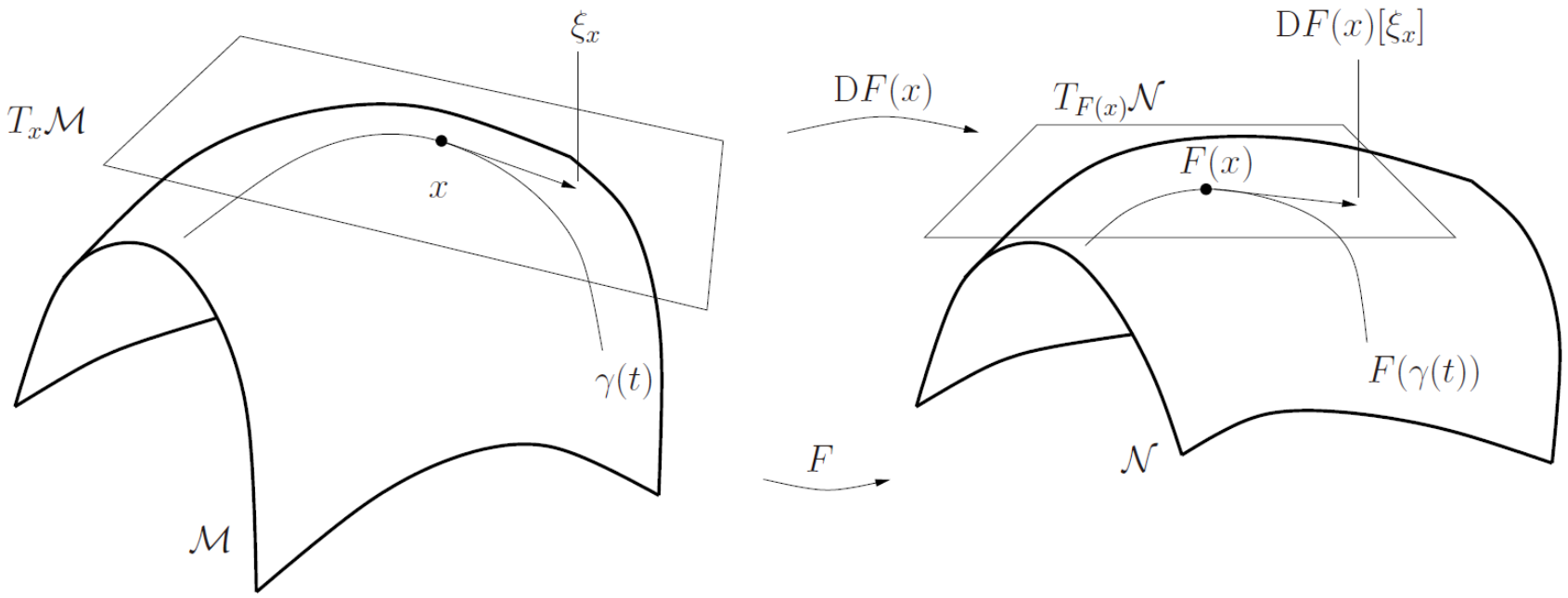
$$d\phi_x: T_x M \to T_{\phi(x)} N$$

- A smooth function f on M can be seen as a map $f: M \to \mathbb{R}$.
  The differential of f at x is an map from $T_x M$ to a real number

$$df_x: T_x M \to \mathbb{R}$$

- If the manifold is $\mathbb{R}^n$, $df_x(\xi)$ is computed by $\nabla f_x^T \xi$, i.e. the inner product of the tangent vector with the gradient.

Differential of a Smooth Map



Construct Manifolds 2: Product Manifolds

- The **product** of an m-dim'l manifold M and n-dim'l manifold N is an $(m+n)$-dim'l manifold $M \times N$.
- At the set level, $M \times N$ consists of all the points of the form (x_1, x_2) where $x_1 \in M$ and $x_2 \in N$.
- $M \times N$ will get a canonical manifold structure.

Examples of Product Manifold

- $\mathbb{R}^n$ can be seen as the product of n real lines $\mathbb{R}$.
- Cylinder $S^1 \times \mathbb{R}$.
- n-Torus $T^n = \underbrace{S^1 \times \cdots \times S^1}_{n}$.

Notes on Product Space

- We are free to product any manifolds M and N to get an product manifold $M \times N$.
- We will encounter the following notation later on:

$$St(r,n) \times GL(r) \times St(r,n)$$

$$\mathcal{O}(r) \times \mathcal{O}(r)$$

Construct Manifolds 3: Quotient Manifolds

- Equivalence relation $\sim$ in the manifold M
  - Reflexive
  - Symmetric
  - Transitive
- Equivalent class $[x] \coloneqq \{y \in M : y \sim x\}$
- If the set $M/\sim$ admit a suitable manifold structure, then it is called a **quotient manifold**.
- The canonical map $\pi : M \to M/\sim$ is called the **canonical projection map**.
- We often define relation $\sim$ using *group actions*.

Quotient of a Group Action

- A **group** G is a set with an operation like addition or multiplication.
- A **Lie group** is a group with manifold structure or manifold with group structure.
- $\mathbb{R}_*$, $O(n)$, $GL(n)$ are typical examples of (Lie) group.
- A Lie group G may **act on a manifold** M

$$\begin{aligned} \sigma: G \times M &\to M \\ (g, x) &\mapsto g \cdot x \end{aligned}$$

- Thus defines a equivalent relation $x \sim y \iff \exists g \in G, s.t.\ y = g \cdot x$
- We denote the **quotient manifold by group action** G (if it is a manifold) by $M/G \coloneqq M/\sim$.
- $\dim(M/G) = \dim M - \dim G$

Example 1 of Quotient Manifold: Space of Circles

- In the punctured plane, we identify circles centered at the origin as equivalent classes, the equivalence relation $\sim$ is defined as

$$x \sim y \iff |x| = |y|$$

- It can also be seen as $\mathbb{R}^2_* / SO(2)$.
- $SO(2)$ is the special orthonormal group, the action on R^2_* is rotation.

Example 2 of Quotient Manifold: Projective Manifolds

- n-dim'l **Real Projective space** RP^n is the set of all lines that pass through origin in $\mathbb{R}^{n+1}$.
- Define the equivalence relation $\sim$ as

$$x \sim y \iff \exists t \in \mathbb{R}, s.t. \, y = tx$$

  Then, $RP^{n-1} = \mathbb{R}_*^n / \sim$.

- RP^1 is the set of all lines in the plane that go through the origin.

- RP^n can be seen as a quotient of group action $RP^n = \mathbb{R}_*^n / \mathbb{R}_*$.
  $t \in R_*$ acts on $x \in R_*^n$ by scalar multiplication $t \cdot x \coloneqq tx$.

Example 3 of Quotient Manifold: Grassmann Manifolds

- A **Grassmann Manifold** $Gr(p,n)$ is the set of all p-dim'l subspace of $\mathbb{R}^n$.
- $Gr(p,n) = \mathbb{R}_*^{n\times p}/GL(p)$ is also an example of the quotient of a group action.
- $Gr(1,n) = RP^{n-1}$.

- $Gr(2,3)$ is all the planes in the 3-dim'l vector space passing through the origin.

- The dimension of $Gr(p,n)$ is $p(n-p)$.

Fixed Rank Matrices as Quotient Manifolds

- The rank-r matrices $\mathbb{R}_r^{n\times m}$ can be characterized by quotient manifolds.
- Factor the matrix $X \in \mathbb{R}_r^{n\times m}$ as

$$X = URV^T$$

where $U \in St(r,n), V \in St(r,m)$ and $R \in R_*^{r\times r}$.

- Since $X = URV^T = (UO_1)O_1^T RO_2(VO_2)^T$ for any $O_1, O_2 \in \mathcal{O}(r)$, we can define group $\mathcal{O}(r)\times\mathcal{O}(r)$ act on the space $St(r,n)\times GL(r)\times St(r,n)$ by

$$\big(\mathcal{O}(r)\times\mathcal{O}(r)\big)\times\big(St(r,n)\times GL(r)\times St(r,n)\big) \to St(r,n)\times GL(r)\times St(r,n)$$

$$\big((O_1,O_2),(U,R,V)\big) \mapsto (UO_1, O_1^T RO_2, VO_2)$$

- So, $\mathbb{R}_r^{n\times m}$ can be seen as a quotient

$$\mathbb{R}_r^{n\times m} = St(r,n)\times GL(r)\times St(r,n)/\mathcal{O}(r)\times\mathcal{O}(r)$$

Riemannian Manifold and Riemannian Metric

- A **Riemannian manifold** (M, g) consists of a smooth manifold M and a **Riemannian Metric** g which provide an inner product g_x on each of the tangent space $T_x M$ of M.
- An **inner product** is a symmetric, positive-definite bilinear form.
- We use inner product and metric synonymously.

- Metric related concepts:
  - Length
  - Volume
  - Angles

Compute Length of a Curve in Riemannian Manifold

- Let $\gamma: [0,1] \to (M, g)$ is a curve in Riemannian manifold (M, g).
- The length of γ is defined as

$$l(\gamma) = \int_0^1 \sqrt{g\big(\dot{\gamma}(t), \dot{\gamma}(t)\big)}dt$$

Notes on Riemannian Manifolds

- A submanifold of a Riemannian Manifold can inherit a canonical Riemannian metric, so it is also a Riemannian manifold.
- The product of two Riemannian manifolds (M, g_1) and (N, g_2) is a Riemannian manifold $(M \times N, g_1 + g_2)$.
- We can also define Riemannian quotient manifolds.

Example 1 of Riemannian Metric: Inner Product of Euclidean Space

- The canonical inner product $g_0(x, y) \coloneqq x^T y$ provide a natural metric on $\mathbb{R}^n$.
- A vector space endowed with the natural inner product $(\mathbb{R}^n, g_0)$ is called **Euclidean Space**.

- Nonstandard inner product is possible. Given $n \times n$ symmetric, positive definite matrix A, define inner product g on $\mathbb{R}^n$ as

$$g(x, y) \coloneqq x^T A y$$

  then, $(\mathbb{R}^n, g)$ is a Riemannian manifold.
- This manifold can be viewed as a stretched version of normal space $\mathbb{R}^n$, for some directions may be longer than others.

Example 2 of Riemannian Metric: Metrics on Matrices

- The most simple and natural inner product one can define on $\mathbb{R}^{m\times n}$ is

$$\langle X, Y\rangle \coloneqq tr(X^T Y)$$

- On $\mathbb{R}^{m\times 1} \cong R^m$, the above inner product is the canonical one on an Euclidean space.
- Frobenius norm is the norm induced by this metric

$$\|A\|_F = \sqrt{tr(A^T A)} = \sqrt{\langle A, A\rangle}$$

- $\mathbb{R}_*^{n\times p}$ and $St(p,n)$ can inherit that metric from $\mathbb{R}^{n\times p}$.

Riemannian Quotient Manifolds

- If we want to define Riemannian structure on the quotient manifold of a Riemannian manifold, we have to assign an inner product to the tangent space of each point.
- We must deal with the problem that on the equivalence classes, different point have different tangent space and different inner products.
- To tackle this problem, we need the following concept of *Horizontal lift*.

Vertical Space and Horizontal Space

- Consider the canonical projection

$$\pi: \bar{M} \to M = \bar{M}/\sim$$

  and a point $x \in M$.
- The equivalence class $\pi^{-1}(x)$ is a manifold.
- Let $\bar{x} \in \bar{M}$ be an element of the equivalence class $\pi^{-1}(x)$.
- The tangent space

$$\mathcal{V}_{\bar{x}} = T_{\bar{x}}\left(\pi^{-1}(x)\right)$$

  is a subspace of $T_{\bar{x}}\bar{M}$, it is called the **vertical space** at $\bar{x}$.
- In Riemannian manifold, the orthogonal complement of vertical space can serve as the **horizontal space**.

$$\mathcal{H}_{\bar{x}} \coloneqq (T_{\bar{x}}\mathcal{V}_{\bar{x}})^{\perp}$$

Horizontal Lift

- The differential of the canonical projection $\pi: \bar{M} \to M$ induces an isomorphism between the horizontal space and the tangent space of the quotient manifold.

$$D\pi(\bar{x}): \mathcal{H}_{\bar{x}} \to T_x M$$

- So, given a tangent vector $\xi_x \in T_x M$, we can assign an unique horizontal tangent vector $\bar{\xi}_{\bar{x}}$ in the tangent space $T_{\bar{x}}\bar{M}$.
- This unique horizontal vector $\bar{\xi}_{\bar{x}}$ is called the **horizontal lift** of ξ_x at $\bar{x}$.
- Horizontal lift gives us a way to 'align' the tangent spaces of points in an equivalence class, so that tangent vectors on the quotient manifold can find unique representatives on the vector spaces of the original manifold.
- This 'alignment' can be used to define Riemannian metric on the quotient space.

Definition of Riemannian Quotient Manifold

- Given Riemannian manifold $(\bar{M}, \bar{g})$ and the canonical projection $\pi: \bar{M} \to M$
- If the metric is consistent along each of the equivalence class, i.e. the inner product $\bar{g}_{\bar{x}}(\bar{\xi}_{\bar{x}}, \bar{\zeta}_{\bar{x}})$ does not depend on $\bar{x} \in \pi^{-1}(x)$, then we can define

$$g_x(\xi_x, \zeta_x) \coloneqq \bar{g}_{\bar{x}}\left(\bar{\xi}_{\bar{x}}, \bar{\zeta}_{\bar{x}}\right)$$

  as the metric of the quotient manifold M.
- Endowed with the above metric, M is called a **Riemannian quotient manifold** of $\bar{M}$.

Example 1 of Riemannian Quotient Manifolds: Space of Circles

- The normal inner product on the punctured plane $\mathbb{R}^2_*$ can be used to define a metric on the quotient space $\mathbb{R}^2_*/SO(2)$.
- The result metric is identical with the normal metric of a real line $\mathbb{R}$.

Example 2 of Riemannian Quotient Manifolds: Projective Space

- For the projective space $RP^{n-1} = \mathbb{R}_*^n / \mathbb{R}_*$, the canonical inner product on $\mathbb{R}_*^n$ does not provide a Riemannian metric for RP^{n-1}.
- Define inner product on $T_y\mathbb{R}_*^n$ as

$$\langle \xi_y, \eta_y \rangle := \frac{1}{y^T y} \xi^T \eta$$

  where $\xi_y, \eta_y \in T_y\mathbb{R}_*^n$.
- This inner product can induce a Riemannian metric on the quotient space RP^{n-1}.

Example 3 of Riemannian Quotient Manifolds: Grassmann Manifolds

- Analogous to the projective space case, define the metric on $\mathbb{R}_*^{n\times p}$ as

$$g_Y(Z,W) = tr\left((X^TX)^{-1}Z^TW\right)$$

  where $Y \in \mathbb{R}_*^{n\times p}$ and $Z, W \in T_Y\mathbb{R}_*^{n\times p}$.

- We can check that this metric can induce to the quotient space $\mathbb{R}_*^{n\times p}/GL(p)$, i.e. the Grassmann manifold $Gr(p,n)$.

Recall of Some Optimization Methods on Euclidean Spaces

- Gradient Descent
- Trust-Region
- Conjugate Gradient
- Newton Method

Unconstrained Optimizations

- We briefly review some classical unconstrained optimization techniques.
- In this section, Euclidean spaces are assumed to be the search space.
- Later on, search space will be generalized to Riemannian manifolds.

- In general, optimization algorithms generate a sequence of points $\{x_k\}_{k=0}^{\infty}$ from the initial point x_0, in the hope that they can converge to the solution fast.
- Different methods differ in the strategy of choosing the next iteration point x_{k+1}.

- Suppose the objective function to be optimized is f.

Gradient Descent Method

- Gradient descent, or steepest descent is a first-order optimization algorithm which belongs to line search strategy.
- **Line search strategies** first choose a descent direction, then choose a proper step length.
- **Gradient descent method** moves along the steepest-descent direction $p_k = -\nabla f_k$ at every step

$$x_{k+1} = x_k + \alpha_k p_k$$

where α_k is a step length needed to be assigned.

Armijo Condition

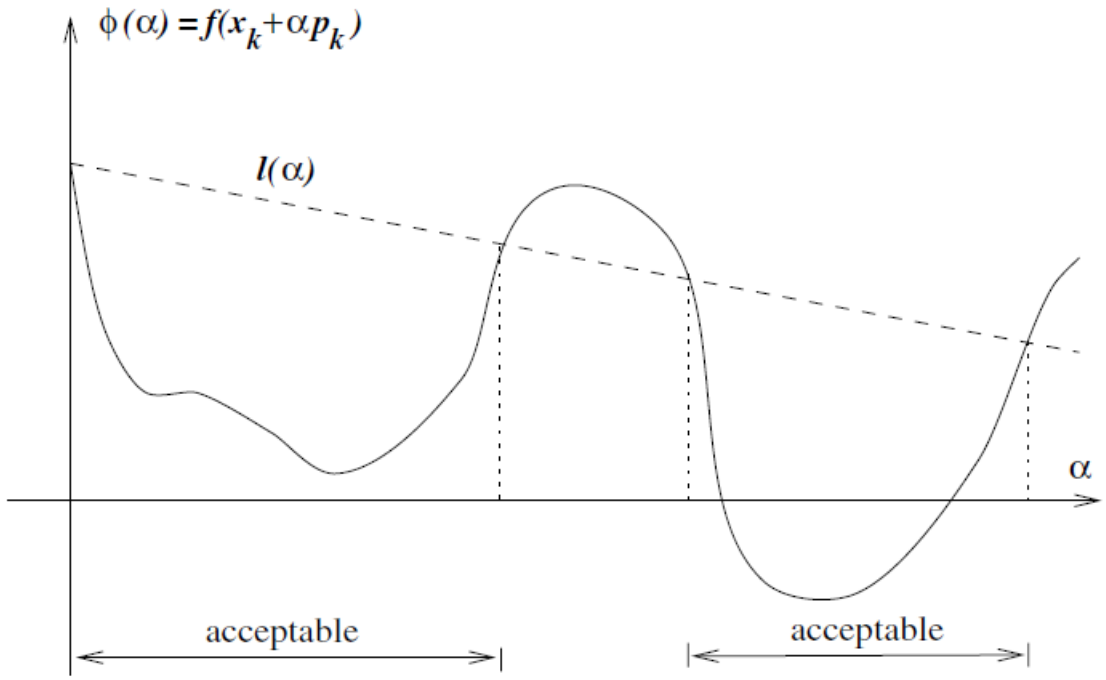
- In choosing the **step length** α_k, we want a sufficient decrease of f.
- The sufficient decrease can be measured by the **Armijo condition**:

$$f(x_k + \alpha p_k) \leq f(x_k) + c_1 \alpha \nabla f_k^T p_k$$

for some parameter $c_1 \in (0,1)$.

- Armijo condition can always be satisfied when α is small.

Armijo Condition



Trust-Region Methods

- **Trust-Region method** generate steps with the help of a quadratic model of f

$$f(x_k + p) \approx m_k(p) = f_k + \nabla f_k^T p + \frac{1}{2} p^T \nabla^2 f_k p \quad s.t. \|p\| \leq \Delta_k$$

  where $\nabla^2 f_k$ is the Hessian.
- Since the quadratic model is an approximation, a region $\|p\| \leq \Delta_k$ that can be trusted should be specified.
- Then choose the step to be the minimizer of the model m_k in this trust region.
- Two sub-problems:
  - Decide the region radius Δ_k,
  - Find the minimizer of m_k in the region.

Choosing the Region Radius Δ_k

- Define the ratio ρ_k between the actual reduction and the predicted reduction

$$\rho_k = \frac{f(x_k) - f(x_k + p_k)}{m_k(0) - m_k(p_k)}$$

- If ρ_k is close to 1, the model is a good approximation of f, so it's safe to increase Δ_k.
- If ρ_k is close to 0 or be negative, we shrink the region. And in the negative case, we should also reject the current step and retry.
- Otherwise, Δ_k stay unchanged.

Minimize in the Region: The Cauchy Point

- Often we find an approximate minimizer of m_k in the trust region rather than the exact solution.
- To achieve global convergence, we need sufficient reduction which can be quantified in terms of the *Cauchy point*.
- **Cauchy point** p_k^C is the minimizer along the gradient direction in the trust region.
- An adapted conjugate gradient method can be used to find an approximate minimizer and guarantees better reduction than the Cauchy point.

Introduction to Conjugate Gradient Methods: Coordinate Descent Method

- An approach that is easy to use is to cycle through the n coordinate directions, minimize f in these directions in turn and repeat after a cycle.
- Simple, intuitive but inefficient.
- The best we can imagine is to cycle just once and get the solution.
- For $f(x) = \frac{1}{2}x^T \Lambda x - b^T x$ where Λ is a diagonal $n \times n$ matrix, it's easy to see coordinate descent method will stop in one cycle, i.e. no more than n steps.
- Surprisingly, for some more general quadratic functions, we can also reach the minimum in n steps---but under suitably chosen coordinate directions.

Conjugate Directions

- Consider minimization of the quadratic function

$$\phi(x) = \frac{1}{2} x^T A x - b^T x$$

  where A is an $n \times n$ symmetric, positive definite matrix.

- A set of directions(vectors) $\{p_0, p_1, \ldots, p_l\}$ is said to be conjugate w.r.t. A if

$$p_i^T A p_j = 0, \qquad \forall i \neq j$$

- The set of conjugate vectors are linearly independent, so a full set of such vectors form a basis to the vector space.

Conjugate Directions as Coordinates

- Given a full set of conjugate directions $\{p_0, p_1, \dots, p_{n-1}\}$ and put them into a matrix $S = [p_0\ p_1 \dots p_{n-1}]$.
- Change the coordinates so that p_i are new coordinate directions

$$x = S\tilde{x}$$

- The objective function ϕ under new coordinates becomes

$$\tilde{\phi}(\tilde{x}) = \phi(S\tilde{x}) = \frac{1}{2}\tilde{x}^T(S^T A S)\tilde{x} - (S^T b)^T\tilde{x}$$

  where the matrix $S^T AS$ is diagonal according to conjugacy.
- If we minimize function f along these new coordinate directions one by one, we will get to the solution in n steps.

Conjugate Direction Method

- In fact, we need not to do the coordinate transformation explicitly. Just optimize along these conjugate directions in turn is fine. This is the **conjugate direction method**.
- Given a starting point x_0 and conjugate directions $\{p_0, p_1, \dots, p_{n-1}\}$, the sequence $\{x_k\}$ is generated by

$$x_{k+1} = x_k + \alpha_k p_k$$

$$\alpha_k = -\frac{\nabla \phi_k^T p_k}{p_k^T A p_k}$$

- Here x_{k+1} is the minimizer of $\phi(x)$ along the line passing x_k with direction p_k.

Conjugate Gradient Method

- The real **conjugate gradient method(CG)** provide a clever way to generate conjugate directions p_k.
- Starting from a gradient direction as p_0, this method can compute a new vector p_{k+1} using only the previous vector p_k while promising the conjugacy of p_{k+1} to all previous p_i $(i \leq k)$.
- The magical code is

$$p_{k+1} = -\nabla\phi_{k+1} + \beta_{k+1}p_k$$

$$\beta_{k+1} = \frac{\nabla\phi_{k+1}^T A p_k}{p_k^T A p_k}$$

- In practice, β_k can be computed using an equivalent but faster expression

$$\beta_{k+1} = \frac{\nabla\phi_{k+1}^T \nabla\phi_{k+1}}{\nabla\phi_k^T \nabla\phi_k}$$

Pseudo Code of CG

Given x_0;

Set $r_0 \leftarrow Ax_0 - b$, $p_0 \leftarrow -r_0$, $k \leftarrow 0$;

while $r_k \neq 0$

$$\alpha_k \leftarrow \frac{r_k^T r_k}{p_k^T A p_k};$$

$$x_{k+1} \leftarrow x_k + \alpha_k p_k;$$

$$r_{k+1} \leftarrow r_k + \alpha_k A p_k;$$

$$\beta_{k+1} \leftarrow \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k};$$

$$p_{k+1} \leftarrow -r_{k+1} + \beta_{k+1} p_k;$$

$$k \leftarrow k + 1;$$

end (while)

Note on Conjugate Directions

- Conjugate directions can be understood as a version of orthogonality.
- Using matrix A, we can define an inner product g on $\mathbb{R}^n$

$$g(x, y) \coloneqq x^T A y$$

- Then in the nonstandard Euclidean space $(\mathbb{R}^n, g)$, two vectors x and y are said to be orthogonal if $g(x, y) = x^T A y = 0$, which coincides with the condition of conjugacy.
- Interesting fact: in space $(\mathbb{R}^n, g)$, the contours of the quadratic function $\phi(x) = \frac{1}{2} x^T A x - b^T x$ are spheres.
- In the viewpoint of this Euclidean space, conjugate direction method is optimizing a circular shaped quadratic function along a set of orthogonal directions.

Nonlinear Conjugate Gradient Methods

- The above conjugate gradient method introduced in fact solves a linear system of equations

$$Ax = b$$

  So it is also called the *linear* conjugate gradient method.
- There are a number of nonlinear versions of CG method to deal with a general nonlinear function f.
- We introduce two variations: the Fletcher-Reeves Method(FR-CG) and the Polak-Ribière Method(PR-CG).

Fletcher-Reeves Method and Polak-Ribière Method

- Fletcher-Reeves method makes two simple changes to the linear CG method
- A line search that identifies an approximate minimum of f along the search direction p_k is needed,
- For the new direction $p_{k+1} = -\nabla\phi(x_{k+1}) + \beta_{k+1}p_k$, β_{k+1} is replaced by β_{k+1}^{FR}

$$\beta_{k+1}^{FR} = \frac{\nabla f_{k+1}^T \nabla f_{k+1}}{\nabla f_k^T \nabla f_k}$$

- Polak-Ribière method modifies the β parameter of FR-CG as

$$\beta_{k+1}^{PR} = \frac{\nabla f_{k+1}^T (\nabla f_{k+1} - \nabla f_k)}{\nabla f_k^T \nabla f_k}$$

- The two methods are identical when f is a strongly convex quadratic function.

Back to Trust-Region Sub-problem

- Sub-problem of minimizing m_k in the trust region can be tackled using CG method. This is Steihaug's approach.
- Steihaug's CG differs from standard CG in that extra stopping criteria are added.
- Like, the algorithm should terminate when p_{k+1} run out of the trust region bound; or stop after a certain steps in case of large dimensions.

Scaling

- **Scaling** is changing scales of variables. Things like changing from meters to millimeters.
- Scaling can be regarded as a way of **preconditioning** which aims at improving the eigenvalue distribution of matrices in question.
- We use scaling here in an extended fashion.

- Scaling has impact on some optimization methods like gradient descent, trust-region, etc.
- Newton method is less affected.

Newton Method

- Newton method is a line search strategy which choose its search direction---Newton direction---utilizing the second-order information of f.
- Like the trust-region method, it uses a local quadratic model

$$f(x_k + p) \approx m_k(p) = f_k + \nabla f_k^T p + \frac{1}{2} p^T \nabla^2 f_k p$$

- Suppose $\nabla^2 f_k$ is positive definite, the Newton direction p_k^N is the vector that points to the minimization of $m_k(p)$ from point x_k.
- By setting the derivative of $m_k(p)$ to zero, we get

$$p_k^N = -\nabla^2 f_k^{-1} \nabla f_k$$

- Suppose f is a quadratic function, it is clear that no matter what the scale is, Newton method will always find the right search direction.

Riemannian Metric and Scaling

- We have considered the quadratic function $\phi(x) = \frac{1}{2}x^T Ax - b^T x$ in the context of nonstandard Euclidean space $(\mathbb{R}^n, g)$, where g is inner product

$$g(x, y) \coloneqq x^T Ay$$

  and saw that the contours of ϕ are perfect circles in this space.

- Suppose f has the following Taylor expansion

$$f(x_k + p) \approx m_k(p) = f_k + \nabla f_k^T p + \frac{1}{2} p^T \nabla^2 f_k p$$

- To make the elliptic contours of $m_k(p)$ more circular, it's reasonable to endow the space with a metric associate with the Hessian $\nabla^2 f_k$.
- Under this metric, the Riemannian gradient(the steepest ascent direction) of f at point x_k is exactly $\nabla^2 f_k^{-1} \nabla f_k$, which agrees with the Newton direction.
- So a properly chosen Riemannian metric can serve as a scaling.

More about Riemannian Geometry

- Geodesics, Exponential Map
- Connection
- Gradient, Hessian
- Parallel Transport

Optimization on Surface

- Suppose an objective function f is defined on a surface. To use classical optimization methods on surface, we need to clarify the following notions
  - Straight lines,
  - The steepest descent direction or gradient,
  - Moving a tangent vector from one point to another,
  - Hessian.
- Riemannian geometry provide corresponding concepts as
  - Geodesics,
  - Riemannian gradient,
  - Parallel transport,
  - and Hessian.

Riemannian Gradient

- Given a smooth function f on a Riemannian manifold (M, g)
- The Riemannian gradient of f at x, denoted by $\nabla f(x)$ is the unique element in $T_x M$ that satisfies

$$g(\nabla f(x), \xi_x) = df_x(\xi_x), \qquad \forall \xi_x \in T_x M$$

- in local coordinates, if the inner product at x is G_x in matrix notation, then

$$\nabla f(x) = G_x^{-1} \operatorname{grad} f(x)$$

here $\operatorname{grad} f(x)$ denotes the Euclidean gradient in $\mathbb{R}^n$.

Note on Riemannian Gradient

- The gradient of $f \in C^\infty(M)$ on a submanifold $M \subset N$ is equal to the projection of the gradient $\tilde{f} \in C^\infty(N)$ onto the tangent space of M. Here $\tilde{f}$ is any smooth extension of f to N.

- Suppose $M = \bar{M}/\sim$ is a quotient manifold, then the horizontal lift of the gradient of $f \in C^\infty(M)$ is equal to the gradient of $\bar{f} \in C^\infty(\bar{M})$. Here $\bar{f}$ is the function induced by f.

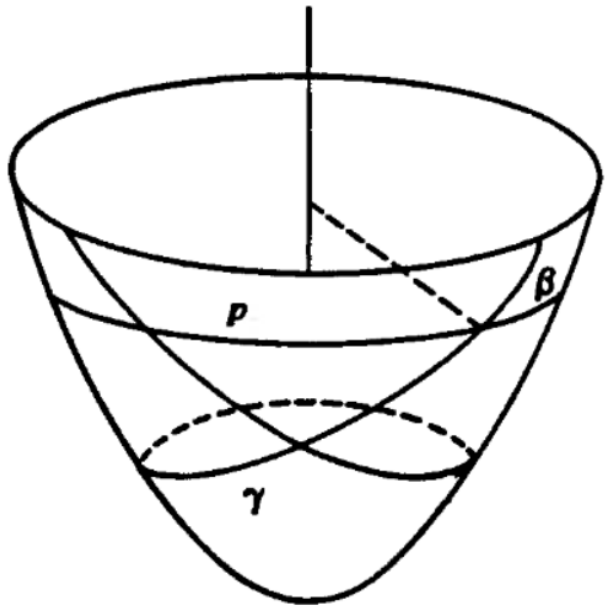
Geodesics

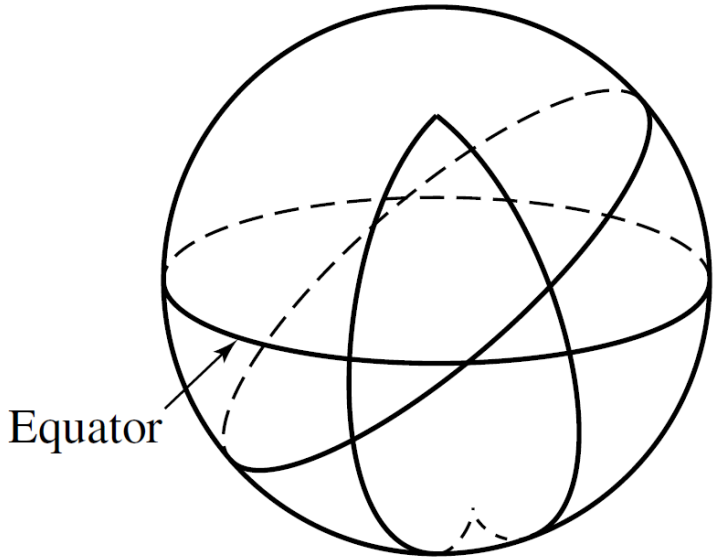
- **Geodesics** is the analogue of straight lines in a curved space.
- Intuitively, geodesic is the shortest path between two points.
- The definition of a geodesic $\gamma: I \to M$ is by a differential equation

$$\ddot{\gamma} = 0$$

which means that curve has a constant speed.

Geodesics

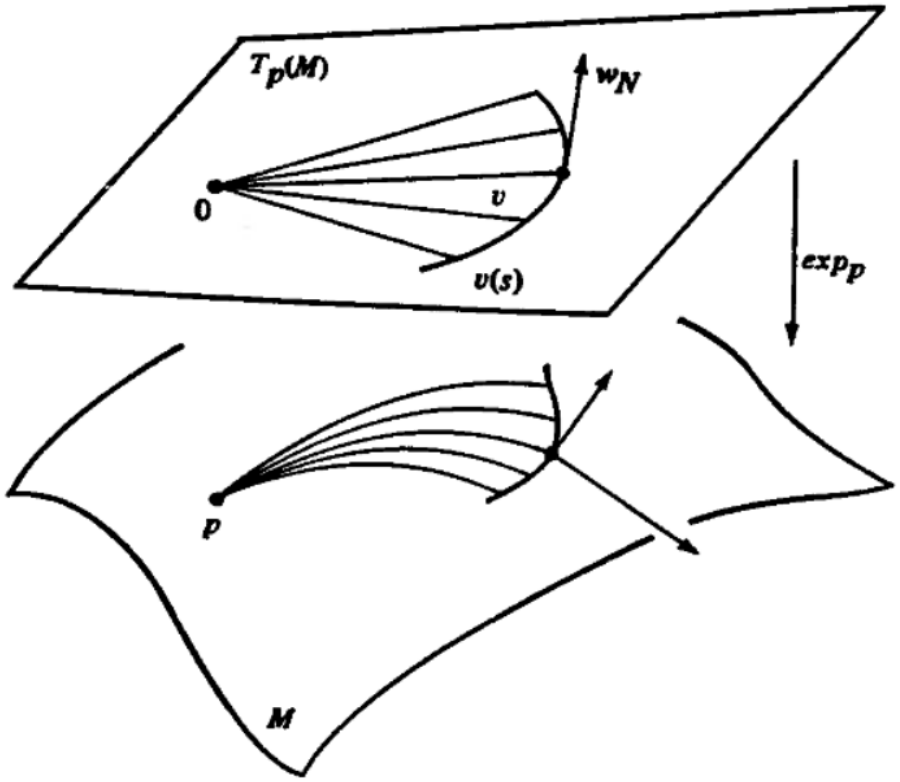




Exponential Map

- **Exponential map** $\exp_x : T_xM \to M$ provide a way to control the surface near a point using the tangent space of that point. This is convenient because tangent space is a vector space while the surface is a curved space.
- Exponential map sends a vector $\xi \in T_xM$ to a point $y \in M$ s.t. y is the endpoint of the unique geodesic $\gamma(t)$ which starts at $\gamma(0) = x$ with initial velocity $\dot{\gamma}(0) = \xi$ and stops at time 1. i.e. $y = \gamma(1)$.
- Generally, exponential map could only be defined locally.

Exponential Map



Connection

- In the definition of geodesics, we compute the acceleration of a curve $\gamma(t)$. In the second derivative, we are differentiating a vector field.
- Unfortunately, neither the manifold structure nor the Riemannian structure provide a natural definition of differentiating a vector field because there are no natural 'connections' between two different vector spaces T_xM and T_yM.
- Connection is an additional structure to differentiate vector fields.
- An **affine connection** ∇ on a manifold M is a mapping

$$\nabla: \mathfrak{X}(M) \times \mathfrak{X}(M) \to \mathfrak{X}(M)$$
$$(\eta, \xi) \mapsto \nabla_\eta \xi$$

  which satisfy several conditions.

Note on Connection

- The vector field $\nabla_\eta \xi$ is called the **covariant derivative** of ξ along η.
- Covariant derivative is a version of directional derivative.
- In $\mathbb{R}^n$, the directional derivative provide a canonical Euclidean connection

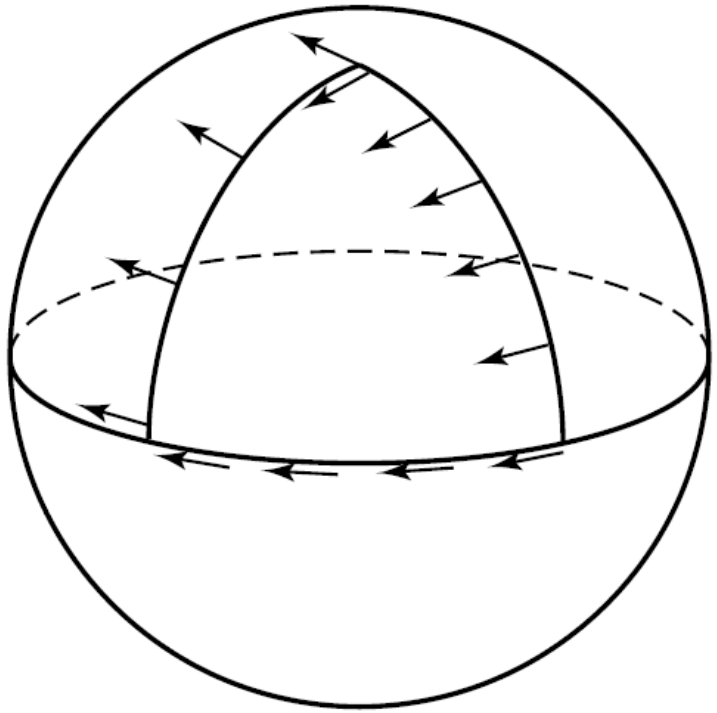
$$\left(\nabla_\eta \xi\right)_x = \lim_{t \to 0} \frac{\xi_{x+t\eta_x} - \xi_x}{t}$$

- Infinitely many connections could be defined on a manifold.
- On Riemannian manifold (M, g), there's one special connection which has good relation with the metric g. It is called the **Levi-Civita connection**.
- The notation of connection ∇ is same as the one of Riemannian gradient. There's no confusion because gradient act on functions while connection or covariant derivative act on vector fields.

Parallel Transport

- In CG methods, we have $p_{k+1} = -\nabla\phi_{k+1} + \beta_{k+1}p_k$.
- $p_k \in T_{x_k}$ wihle $\nabla\phi_{k+1} \in T_{x_{k+1}}$. They belong to different tangent spaces. To do the addition, we need to move the tangent vector p_k at x_k to x_{k+1}.
- A vector field ξ along a curve γ is said to be **parallel along γ** if $\nabla_{\dot{\gamma}}\xi \equiv 0$.
- If ξ, η are two parallel fields along γ, then $g(\xi, \eta)$ is constant along γ.
- So, parallel fields along a curve neither change their lengths nor their angles relative to each other, just as parallel fields in Euclidean space are.

Parallel Transport



Hessian

- In $\mathbb{R}^n$, $B = \text{Hess } f_x$ is a symmetric matrix which can be seen as a function

$$\text{Hess } f_x : T_x\mathbb{R}^n \times T_x\mathbb{R}^n \to \mathbb{R}$$
$$(\xi_x, \eta_x) \mapsto \xi_x^T B \eta_x$$

- Hessian in Riemannian manifold (M, g) is defined as a map

$$\text{Hess } f_x : T_xM \times T_xM \to \mathbb{R}$$
$$(\xi_x, \eta_x) \mapsto g\left(\nabla_{\xi_x} \nabla f, \eta_x\right)$$

- $\text{Hess } f_x(\xi_x, \eta_x)$ can be seen as the component of the directional derivative of ∇f along the tangent vector ξ_x projecting to the direction of η_x.

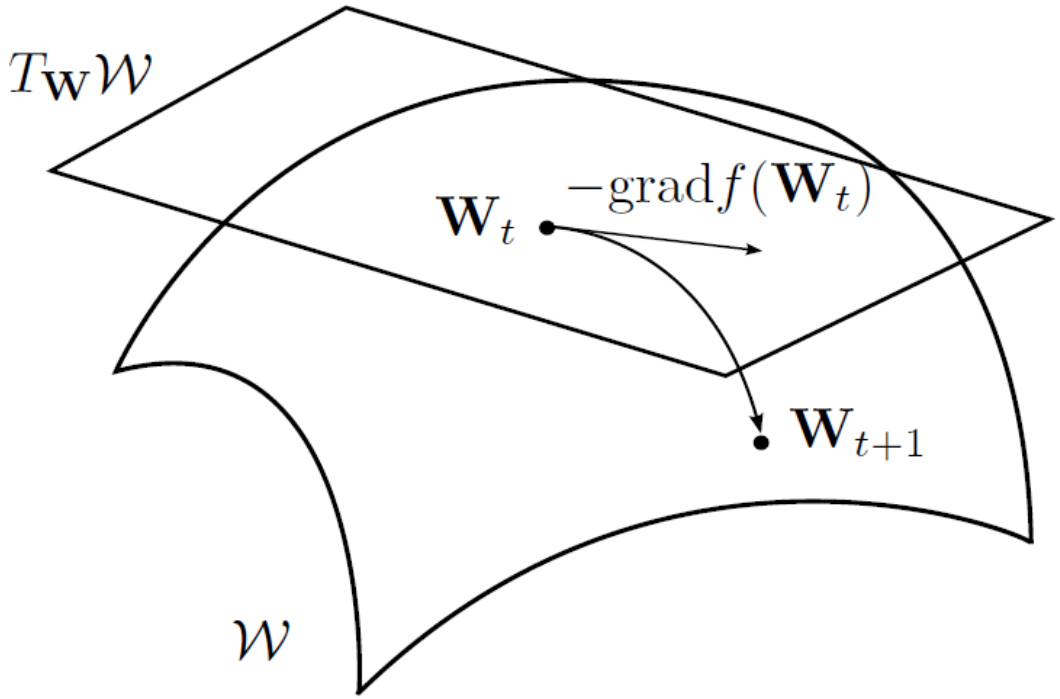
Optimization Related Ingredients of Riemannian Manifolds

- Retraction
- Vector transport

Optimization on Riemannian Manifold

- By replacing moving toward a direction by moving along geodesics and moving vectors by parallel translation, most of the traditional optimization methods can be applied in the manifold context.
- Computing the exact geodesics and parallel translation can be computationally expensive. In practice, we use related concepts such as retraction and vector transport instead.

Gradient Descent on Riemannian Manifold

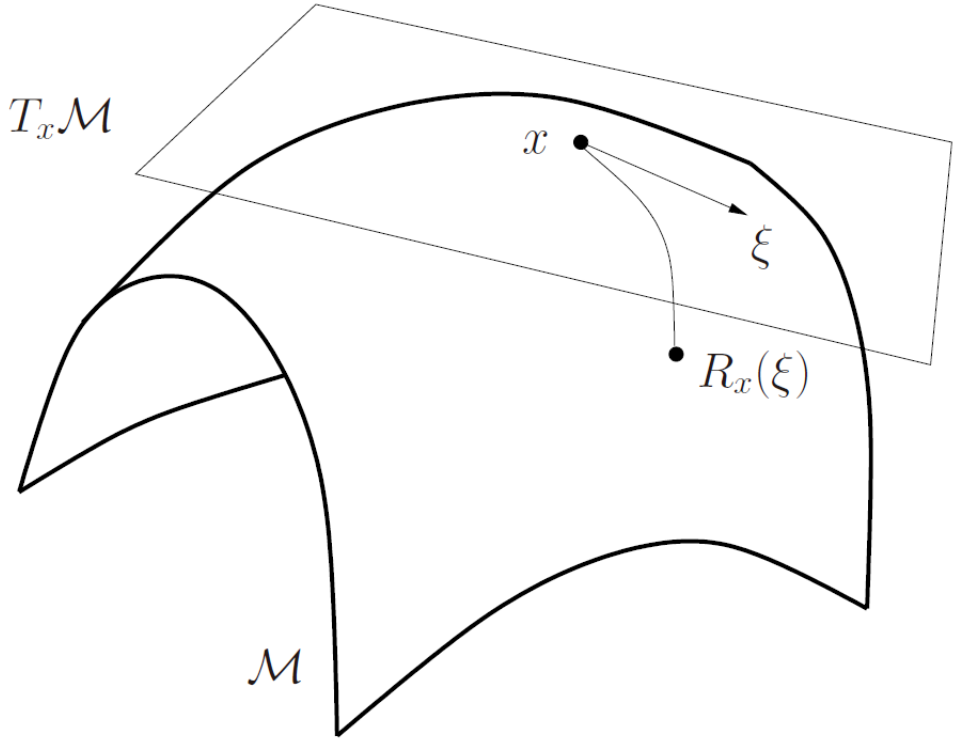


Retraction

- A retraction at $x \in M$ on a manifold M is a smooth mapping $R_x: T_xM \to M$ s.t.
- (1) $R_x(0_x) = x$, where $0_x \in T_xM$,
- (2) $DR_x(0_x) = id_{T_xM}$.

- For embedded submanifold M of $\mathbb{R}^n$, $R_x(\xi)$ is often defined by "projecting" the point $x + \xi$ back to M.
- This "projecting" can be based on finding the nearest point from $x + \xi$ or on matrix decompositions such as QR factorization.

Retraction



Note on Retraction

- For Riemannian manifolds, the exponential map is a retraction.

- In topology, a retraction is a continuous mapping from the entire space into a subspace which preserves the position of all points in that subspace.
- The Retraction we use here is different from the above one. It is better described as numerical version of exponential mapping.

Vector Transport

- A vector transport $\mathcal{T}$ at $x \in M$ associated with retraction R is a smooth mapping

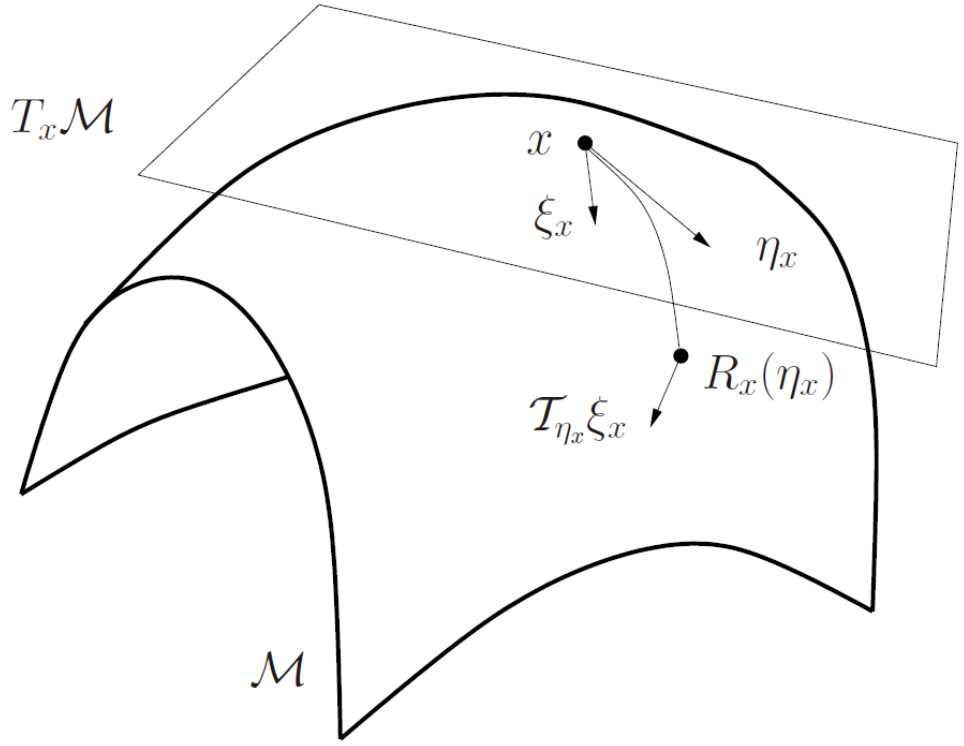
$$\mathcal{T}: TM \oplus TM \to TM$$
$$(\eta_x, \xi_x) \mapsto \mathcal{T}_{\eta_x}(\xi_x)$$

  s.t.

- (1) (associated retraction) $\pi\left(\mathcal{T}_{\eta_x}(\xi_x)\right) = R(\eta_x)$,
- (2) (consistency) $\mathcal{T}_{0_x}(\xi_x) = \xi_x$,
- (3) (linearity) $\mathcal{T}_{\eta_x}(a\xi_x + b\zeta_x) = a\mathcal{T}_{\eta_x}(\xi_x) + b\mathcal{T}_{\eta_x}(\zeta_x)$.

- Parallel translation is a particular vector transport.

Vector Transport



Geometric Line-Search Methods

Input: $f : \mathcal{M} \to \mathbb{R}$, $x_0 \in \mathcal{M}$, $k = 0$
  repeat
    choose a descent direction $p_k \in T_{x_k}\mathcal{M}$
    choose a retraction $R_{x_k} : T_{x_k}\mathcal{M} \to \mathcal{M}$
    choose a step length $\alpha_k \in \mathbb{R}$
    set $x_{k+1} = R_{x_k}(\alpha_k p_k)$
    $k \leftarrow k + 1$
  until x_{k+1} sufficiently minimizes f

Geometric Conjugate Gradient Method

Require: Riemannian manifold $\mathcal{M}$; vector transport $\mathcal{T}$ on $\mathcal{M}$ with associated retraction R; real-valued function f on $\mathcal{M}$.
Goal: Find a local minimizer of f.
Input: Initial iterate $x_0 \in \mathcal{M}$.
Output: Sequence of iterates $\{x_k\}$.

1: Set $\eta_0 = -\operatorname{grad} f(x_0)$.
2: **for** $k = 0, 1, 2, \ldots$ **do**
3: Compute a step size α_k and set

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

4: Compute β_{k+1} and set

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

5: **end for**

Geometric Algorithms for Matrix Completion

- LRGeom
- R3MC
- LMaFit

MC

- The Matrix Completion problem:

$$\arg\min_{X \in \mathbb{R}_r^{n \times m}} f(X) \coloneqq \frac{1}{2} \|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(X^*)\|_F^2$$

- $\mathbb{R}_r^{n \times m}$ is the set of rank-r $n \times m$matrices, the function

$$\mathcal{P}_\Omega(X)_{i,j} = \begin{cases} X_{i,j} & \text{if } (i,j) \in \Omega \\ 0 & \text{otherwise} \end{cases}$$

LRGeom

- LRGeom minimize f on the manifold $\mathbb{R}_r^{n\times m}$ which is seen as a submanifold of $\mathbb{R}^{n\times m}$ using the geometric CG method.

- It is described in [Vandereycken, Bart. "Low-rank matrix completion by Riemannian optimization---extended version." *arXiv:1209.3834* (2012).]

LRGeom

Require: initial iterate $X_1 \in \mathcal{M}_k$, tolerance $\tau > 0$, tangent vector $\eta_0 = 0$

1: **for** $i = 1, 2, \ldots$ **do**

2: Compute the gradient

$\xi_i := \operatorname{grad} f(X_i)$

3: Check convergence

if $\|\xi_i\| \leq \tau$ then break

4: Compute a conjugate direction by PR+

$\eta_i := -\xi_i + \beta_i \mathcal{T}_{X_{i-1} \to X_i}(\eta_{i-1})$

5: Determine an initial step t_i from the linearized problem

$t_i = \arg\min_t f(X_i + t\,\eta_i)$

6: Perform Armijo backtracking to find the smallest integer $m \geq 0$ such that

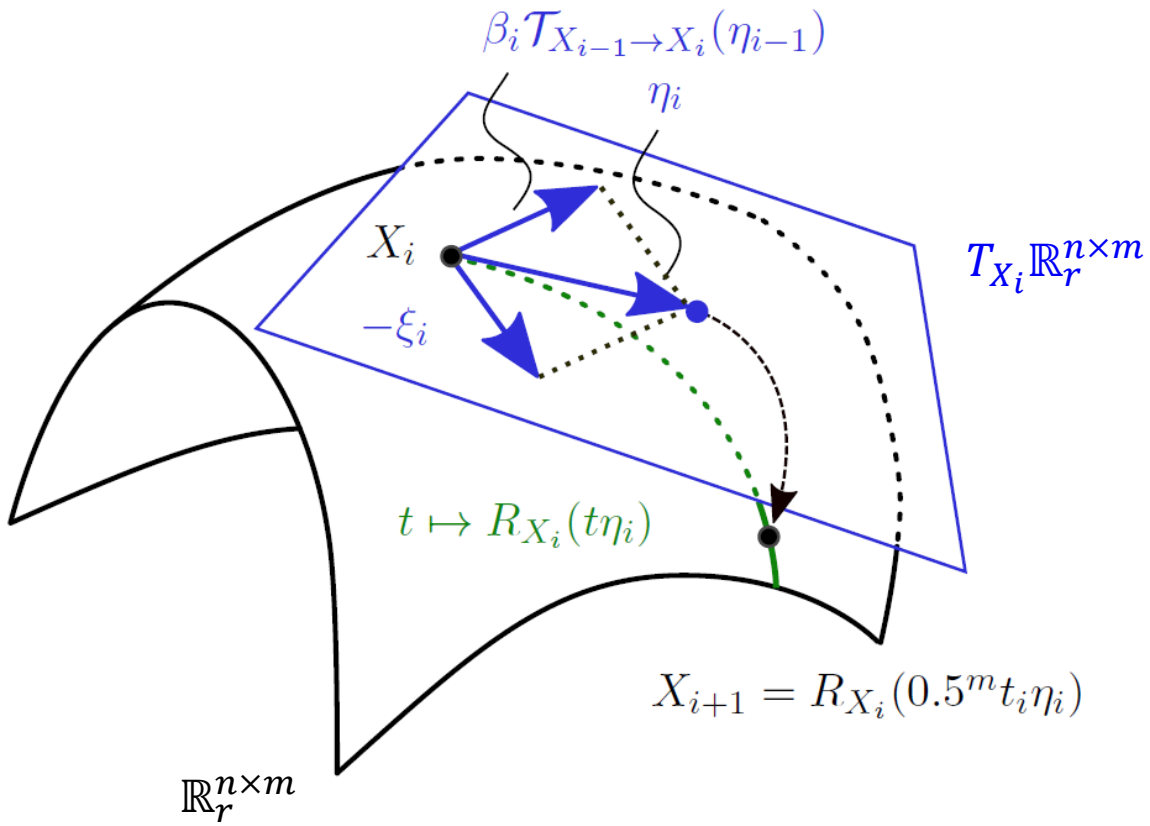
$f(X_i) - f(R_{X_i}(0.5^m\, t_i\, \eta_i)) \geq -0.0001 \times 0.5^m\, t_i \,\langle\, \xi_i, \eta_i \,\rangle$

and obtain the new iterate

$X_{i+1} := R_{X_i}(0.5^m\, t_i\, \eta_i)$

7: **end for**

LRGeom


$\beta_i \mathcal{T}_{X_{i-1} \to X_i}(\eta_{i-1})$
η_i
X_i
$-\xi_i$
$T_{X_i}\mathbb{R}_r^{n\times m}$
$t \mapsto R_{X_i}(t\eta_i)$
$X_{i+1} = R_{X_i}(0.5^m t_i \eta_i)$
$\mathbb{R}_r^{n\times m}$

LRGeom
Tangent Space

- Using SVD,

  $M = \mathbb{R}_r^{n\times m} = \{U\Sigma V^T : U \in St(r,n), V \in St(r,m), \Sigma = diag(\sigma_i), \sigma_1 \geq \cdots \geq \sigma_r > 0\}$

- The tangent space $T_X M$ at $X = U\Sigma V^T \in M$ is

$$T_X M = \left\{ [U\ U_\perp] \begin{bmatrix} \mathbb{R}^{r\times r} & \mathbb{R}^{r\times(m-r)} \\ \mathbb{R}^{(n-r)\times r} & 0 \end{bmatrix} [V\ V_\perp]^T \right\}$$

$$= \left\{ UMV^T + U_p V^T + UV_p^T : \begin{matrix} M \in \mathbb{R}^{r\times r}, \\ U_p \in \mathbb{R}^{n\times r}, U_p^T U = 0, \\ V_p \in \mathbb{R}^{m\times r}, V_p^T V = 0 \end{matrix} \right\}$$

where $U_\perp$ is a matrix s.t. $[U\ U_\perp] \in \mathcal{O}(n)$.

LRGeom Gradient

- The metric g on M is the restriction of the Euclidean inner product on $\mathbb{R}^{n\times m}$

$$g_X(\xi,\eta) = tr(\xi^T\eta), \qquad X \in M \text{ and } \xi,\eta \in T_XM$$

- The Riemannian gradient is the orthogonal projection of the gradient of f seen as a function on $\mathbb{R}^{n\times m}$ onto the tangent space of M

$$\nabla f(X) = P_{T_XM}\big(\mathcal{P}_\Omega(X-A)\big)$$

- P_{T_XM} is the orthogonal projection onto the tangent space at X

$$P_{T_XM}\colon \mathbb{R}^{n\times m} \to T_XM$$
$$Z \mapsto P_U Z P_V + P_U^\perp Z P_V + P_U Z P_V^\perp$$

where $P_U \coloneqq UU^T$ and $P_U^\perp \coloneqq 1 - P_U$.

LRGeom Retraction

- Retraction is based on the orthogonal projection P_M

$$R_X : T_X M \to M$$
$$\xi \mapsto P_M(X + \xi)$$

  where $P_M(Y) \coloneqq \arg\min_{Z \in M} \|Y - Z\|_F$.

- This retraction is well-defined locally.
- $P_M(Y)$ can be computed using SVD. Given $Y = U\Sigma V^T$

$$P_M(Y) = \Sigma_{i=1}^{r} \sigma_i u_i v_i^T$$

LRGeom
Vector Transport

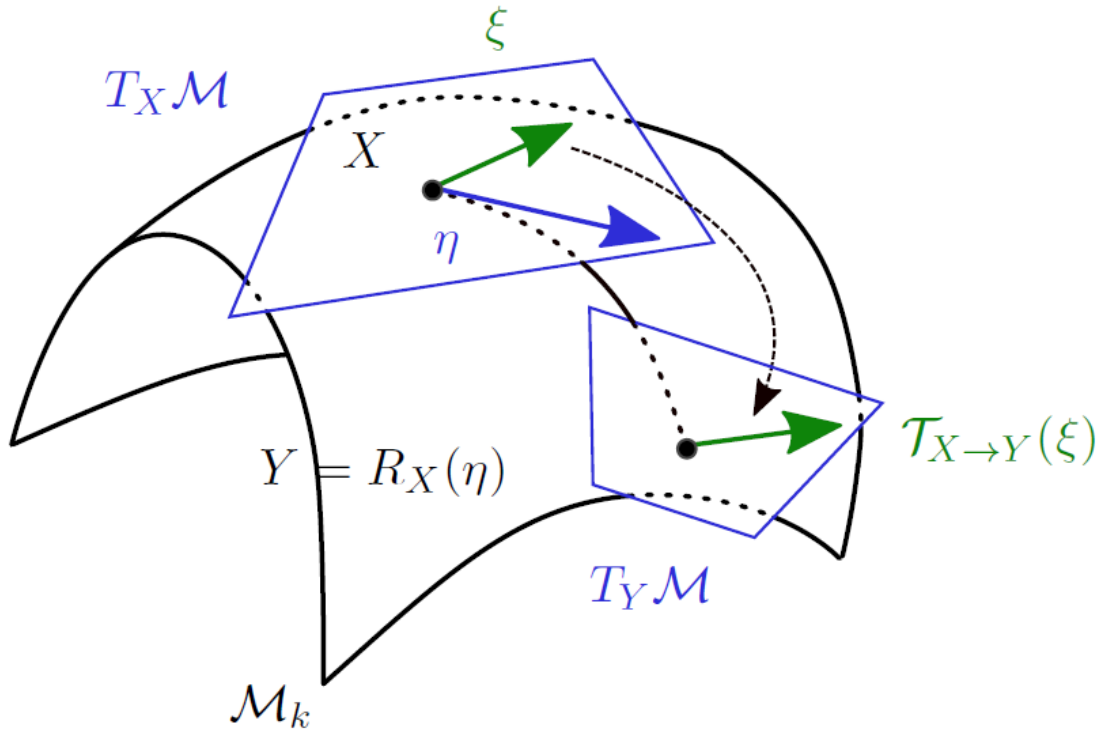
- Introduce a shorthand notation for vector transport

$$\mathcal{T}_{X\to Y}: T_X M \to T_Y M$$
$$\xi \mapsto \mathcal{T}_{R_X^{-1}(Y)}(\xi)$$

- Vector transport is defined by orthogonally projecting the translated tangent vector in $\mathbb{R}^{n\times m}$

$$\mathcal{T}_{X\to Y}: T_X M \to T_Y M$$
$$\xi \mapsto P_{T_Y M}(\xi)$$

LRGeom
Vector Transport



LRGeom
Conjugate Direction

- The conjugate search direction η_i is

$$\eta_i = -\nabla f(X_i) + \beta_i \mathcal{T}_{X_{i-1} \to X_i}(\eta_{i-1})$$

- β_i is chosen by the Polak-Ribière Method

$$\beta_i = \frac{\langle \nabla f(X_i), \nabla f(X_i) - \mathcal{T}_{X_{i-1} \to X_i}(\nabla f(X_{i-1})) \rangle}{\langle \nabla f(X_{i-1}), \nabla f(X_{i-1}) \rangle}$$

LRGeom
Line Search

- The line search sub-procedure of the CG method uses the following value t_* as the initial guess for the step size.

$$t_* \coloneqq \arg\min_t f(X + t\eta) = \frac{1}{2}\arg\min_t \|\mathcal{P}_\Omega(X - X^*) + t\mathcal{P}_\Omega(\eta)\|_F^2$$

- It has a closed-form solution

$$t_* = \frac{\langle \mathcal{P}_\Omega(\eta), \mathcal{P}_\Omega(A - X)\rangle}{\langle \mathcal{P}_\Omega(\eta), \mathcal{P}_\Omega(\eta)\rangle}$$

R3MC

- R3MC is also a Geometric CG method.
- Motivated by the 3-factor factorization $X = URV^T$, R3MC optimize in the quotient space

$$M \coloneqq \mathbb{R}_r^{n\times m} = St(r,n) \times GL(r) \times St(r,n)/\mathcal{O}(r) \times \mathcal{O}(r)$$

  where $\mathcal{O}(r) \times \mathcal{O}(r)$ act on $\bar{M} \coloneqq St(r,n) \times GL(r) \times St(r,n)$ as

$$\big(\mathcal{O}(r) \times \mathcal{O}(r)\big) \times \bar{M} \to \bar{M}$$

$$\big((O_1, O_2), (U, R, V)\big) \mapsto (UO_1, O_1^T R O_2, VO_2)$$

- R3MC is defined in [Mishra, B., and R. Sepulchre. "R3MC: A Riemannian Three-Factor Algorithm for Low-Rank Matrix Completion." (2013).]

R3MC

Input: Initial iterate $\bar{x}_1 = (\mathbf{U}_1, \mathbf{R}_1, \mathbf{V}_1) \in \overline{\mathcal{M}}$, tolerance $\tau > 0$, horizontal vector $\bar{\eta}_0 = 0$

Ouput: Sequence of iterates $\{\bar{x}_i\}$

1: **for** $i = 1, 2, \ldots$ **do**

2: Compute the Riemannian gradient $\bar{\xi}_i = \overline{\text{grad}}_{\bar{x}_i} \bar{\phi}$

3: **if** Check convergence, if $\bar{g}_{\bar{x}_i}(\bar{\xi}_i, \bar{\xi}_i) \leq \tau$ **then**

break

4: Compute a conjugate direction by Polak-Ribière (PR+)

$\bar{\eta}_i = -\bar{\xi}_i + \beta_i(\overline{\mathcal{T}_{\bar{x}_i \leftarrow \bar{x}_{i-1}} \bar{\eta}_{i-1}})_{\bar{x}_i}$

5: **if** $(\bar{g}_{\bar{x}_i}(\bar{\eta}_i, \bar{\xi}_i) > 0)$ **then**

$\bar{\eta}_i = -\bar{\xi}_i$

6: Determine an initial step s_i from the linearized problem

7: Find the smallest integer $p \geq 0$ such that

$\bar{\phi}(\bar{x}_i) - \bar{\phi}(R_{\bar{x}_i}(\frac{s_i}{2^p}\bar{\eta}_i)) \geq \frac{-0.0001}{2^p} s_i \bar{g}_{\bar{x}_i}(\bar{\eta}_i, \bar{\xi}_i)$ and

set $\bar{x}_{i+1} = R_{\bar{x}_i}(\frac{s_i}{2^p}\bar{\eta}_i)$

8: **end for**

R3MC
Conceptual Schema

- Let $\mathcal{E} \coloneqq \mathbb{R}^{n\times r} \times \mathbb{R}^{r\times r} \times \mathbb{R}^{m\times r}$, then we have

$$(\mathcal{E}, \bar{g}) \underset{\supset}{\overset{\text{Riemannian submanifold}}{}} (\bar{M}, \bar{g}) \xrightarrow{\text{Riemannian quotient}} (M, g)$$

- This view of the search space allows us to derive various notions on the Riemannian quotient manifold M in a systematic way.

R3MC
Tangent Space

- We represent an element x of the quotient space M by $x = [\bar{x}]$ where $\bar{x} \in \bar{M}$ and has matrix representation $\bar{x} = (U, R, V)$.
- Tangent space of the total space $\bar{M}$ at $\bar{x}$

$$T_{\bar{x}}\bar{M} = \left\{ (Z_U, Z_R, Z_V) \in \mathbb{R}^{n\times r} \times \mathbb{R}^{r\times r} \times \mathbb{R}^{m\times r} : \begin{matrix} U^T Z_U + Z_U^T U = 0 \\ V^T Z_V + Z_V^T V = 0 \end{matrix} \right\}$$

R3MC Metric

- The metric $\bar{g}$ on $T_{\bar{x}}\bar{M}$ is

$$\bar{g}_{\bar{x}}(\bar{\xi}_{\bar{x}}, \bar{\eta}_{\bar{x}}) = tr\left((RR^T)\bar{\xi}_U^T\bar{\eta}_U\right) + tr\left(\bar{\xi}_R^T\bar{\eta}_R\right) + tr\left((R^TR)\bar{\xi}_V^T\bar{\eta}_V\right)$$

- The metric g on the quotient space M is induced by $\bar{g}$.

- This metric captures the characterization of the cost function f which leads to good preconditioning properties.
- In fact, this metric is induced by the Euclidean Hessian of f (using a diagonal approximation of the full Hessian).

R3MC
Vertical Space

- The vertical space $\mathcal{V}_{\bar{x}}\bar{M}$ of $T_{\bar{x}}\bar{M}$ at $\bar{x} = (U, R, V)$ has the form

$$\mathcal{V}_{\bar{x}}\bar{M} = \left\{ (U\Omega_1, R\Omega_2 - \Omega_1 R, V\Omega_2) : \begin{matrix} \Omega_1^T = -\Omega_1 \\ \Omega_2^T = -\Omega_2 \end{matrix} \right\}$$

R3MC
Horizontal Space

- The horizontal space $\mathcal{H}_{\bar{x}}\overline{M}$ of $T_{\bar{x}}\overline{M}$ at $\bar{x} = (U, R, V)$ has the form

$$\mathcal{H}_{\bar{x}}\overline{M} = \left\{ \bar{\eta}_{\bar{x}} = (\bar{\eta}_U, \bar{\eta}_R, \bar{\eta}_U) \in T_{\bar{x}}\overline{M} : \begin{array}{l} RR^T U^T \bar{\eta}_U - \bar{\eta}_R R^T \text{ is symmetric} \\ R^T R V^T \bar{\eta}_V + R^T \bar{\eta}_R \text{ is symmetric} \end{array} \right\}$$

- It is computed by the condition that $\mathcal{H}_{\bar{x}}\overline{M} = (\mathcal{V}_{\bar{x}}\overline{M})^{\perp}$.

R3MC
Horizontal Space

- Let $\bar{x} = (U, R, V)$, $\bar{\eta}_{\bar{x}} = (\bar{\eta}_U, \bar{\eta}_R, \bar{\eta}_V) \in \mathcal{H}_{\bar{x}}\bar{M}$, $\bar{\xi}_{\bar{x}} = (U\Omega_1, R\Omega_2, V\Omega_2) \in \mathcal{V}_{\bar{x}}\bar{M}$, then

$$\bar{g}_{\bar{x}}\left(\bar{\eta}_{\bar{x}}, \bar{\xi}_{\bar{x}}\right) = tr(R^T R\bar{\eta}_U^T U\Omega_1) + tr(\Omega_2^T R^T \bar{\eta}_R - R^T \Omega_1^T \bar{\eta}_R) + tr(R^T R\bar{\eta}_V^T V\Omega_2)$$

$$= tr(-R^T RU^T \bar{\eta}_U \Omega_1) + tr(\bar{\eta}_R R^T \Omega_1) + tr(-R^T RV^T \bar{\eta}_V \Omega_2) + tr(-R^T \bar{\eta}_R \Omega_2)$$

$$= tr\left((-R^T RU^T \bar{\eta}_U \Omega_1 + \bar{\eta}_R R^T)\Omega_1\right) + tr\left((-R^T RV^T \bar{\eta}_V - R^T \bar{\eta}_R)\Omega_2\right) = 0$$

$$= 0$$

- Then $-R^T RU^T \bar{\eta}_U \Omega_1 + \bar{\eta}_R R^T$ and $-R^T RV^T \bar{\eta}_V - R^T \bar{\eta}_R$ must be symmetric.

R3MC
Projection Ψ

- Since $(\bar{M}, \bar{g}) \subset (\mathcal{E}, \bar{g})$, we define the projection operator $\Psi_{\bar{x}}$ which project $T_{\bar{x}}\mathcal{E} = \mathcal{E}$ onto $T_{\bar{x}}\bar{M}$.
- This can be done by subtracting the normal component. The normal space

$$N_{\bar{x}}\bar{M} = \left\{ (UN_1, 0, VN_2): \begin{matrix} N_1, N_2 \in \mathbb{R}^{r\times r} \\ N_1RR^T \text{ and } N_2R^TR \text{ are symmetric} \end{matrix} \right\}$$

- So,

$$\Psi_{\bar{x}}: \mathcal{E} \to T_{\bar{x}}\bar{M}$$
$$(Z_U, Z_R, Z_V) \mapsto (Z_U - UB_U(RR^T)^{-1}, Z_R, Z_V - VB_V(R^TR)^{-1})$$

- Here B_U and B_V are $r \times r$ symmetric matrices which satisfies the Lyapunov Equations

$$RR^TB_U + B_URR^T = RR^T(U^TZ_U + Z_U^TU)RR^T$$
$$R^TRB_V + B_VR^TR = R^TR(V^TZ_V + Z_V^TV)RR^T$$

R3MC
Projection Ψ

- Let $Z_U = UN_1(\text{normal part}) + T(\text{tangent part})$.
- Let $B_U \coloneqq N_1(RR^T)$, $T = U\Omega + U_\perp K$ where B_U is symmetric and Ω anti-symmetric.
- Then

$$\begin{aligned} Z_U &= UN_1 + T \\ &= U(B_U + \Omega RR^T)(RR^T)^{-1} + U_2 K \end{aligned}$$

- So, $U^T Z_U = (B_U + \Omega RR^T)(RR^T)^{-1}$ and $Z_U^T U = (RR^T)^{-1}(B_U - RR^T\Omega)$. We get the Lyapunov equations

$$RR^T B_U + B_U RR^T = RR^T(U^T Z_U + Z_U^T U)RR^T$$
$$R^T R B_V + B_V R^T R = R^T R(V^T Z_V + Z_V^T V)RR^T$$

R3MC
Lyapunov Equations

- Lyapunov equation

$$RR^T B_U + B_U RR^T = E$$

can be solved efficiently by diagonalizing R(using SVD).

R3MC
Projection Π

- The projection onto the horizontal space is done by the operator Π

$$\Pi_{\bar{x}}: T_{\bar{x}}\bar{M} \to \mathcal{H}_{\bar{x}}\bar{M}$$
$$\bar{\xi}_{\bar{x}} \mapsto \left(\bar{\xi}_U - U\Omega_1, \bar{\xi}_R + \Omega_1 R - R\Omega_2, \bar{\xi}_V - V\Omega_2\right)$$

- Here Ω_1 and Ω_2 are $r \times r$ skew symmetric matrices which satisfies the Lyapunov equations

$$RR^T\Omega_1 + \Omega_1 RR^T - R\Omega_2 R^T = \mathrm{Skew}\left(U^T\bar{\xi}_U RR^T\right) + \mathrm{Skew}\left(R\bar{\xi}_R^T\right)$$
$$R^T R\Omega_2 + \Omega_2 R^T R - R^T\Omega_1 R = \mathrm{Skew}\left(V^T\bar{\xi}_V R^T R\right) + \mathrm{Skew}\left(R^T\bar{\xi}_R\right)$$

R3MC
Gradient

- Let $\bar{f}(\bar{x}) \coloneqq \frac{1}{2}\|\mathcal{P}_\Omega(URV^T) - \mathcal{P}_\Omega(X^*)\|_F^2$ be the cost function on $(\bar{M}, \bar{g})$ and f be its induced function on the quotient manifold (M, g).
- Denote $S \coloneqq \mathcal{P}_\Omega(URV^T) - \mathcal{P}_\Omega(X^*)$, the gradient $\nabla_{\bar{x}}\bar{f}$ of $\bar{f}$ on $(\mathcal{E}, \bar{g})$ can be written in terms of S

$$\nabla_{\bar{x}}\bar{f} = (SVR^T(RR^T)^{-1}, U^TSV, S^TUR(R^TR)^{-1})$$

- The horizontal lift of $\nabla_x f$ is equal to the gradient of $\bar{f}$ on $(\bar{M}, \bar{g})$ which is the horizontal projection of gradient of $\bar{f}$ on $(\mathcal{E}, \bar{g})$

$$\overline{\nabla_x f} = \Psi_{\bar{x}}\left(\nabla_{\bar{x}}\bar{f}\right)$$

R3MC
Retraction

- The retraction here is obtained by matrix factorization

$$R_{\bar{x}}(\bar{\xi}_{\bar{x}}) = \left(\mathrm{qf}(U + \bar{\xi}_U), R + \bar{\xi}_R, \mathrm{qf}(V + \bar{\xi}_V)\right)$$

- Here, $\bar{\xi}_{\bar{x}} \in \mathcal{H}_{\bar{x}}\bar{M}$ and uf is the orthogonal part of a QR decomposition.
- $\mathrm{uf}(A)$ can be computed efficiently by performing the SVD of A.

R3MC
Vector Transport

- Vector transport here is similar to the one of LRGeom.
- The horizontal lift of the vector transport

$$\overline{\mathcal{T}_{x\to y}}: \mathcal{H}_{\bar{x}}\bar{M} \to \mathcal{H}_{\bar{y}}\bar{M}$$
$$\bar{\xi}_{\bar{x}} \mapsto \Pi_{\bar{y}}\left(\Psi_{\bar{y}}\left(\bar{\xi}_{\bar{x}}\right)\right)$$

R3MC
Line Search

- The initial guess t_* for the step size of the line search sub-procedure is obtained by solving the min problem

$$t_* = \arg\min_t \left\| \mathcal{P}_\Omega\left((U - t\bar{\xi}_U)(R - t\bar{\xi}_R)(V - t\bar{\xi}_V)\right) - \mathcal{P}_\Omega(X^*) \right\|_F^2$$

- It can be accelerated by computing a degree 2 polynomial approximation

$$t_*^{\text{accel}} = \arg\min_t \left\| \mathcal{P}_\Omega\left(URV^T - t\left(\bar{\xi}_U RV^T + U\bar{\xi}_R V^T + UR\bar{\xi}_V\right)\right) - \mathcal{P}_\Omega(X^*) \right\|_F^2$$

  which has a closed form solution.

LMaFit

- LMaFit---a **L**ow-rank **M**atrix **Fit**ting algorithm.
- It is described in [Wen, Zaiwen, Wotao Yin, and Yin Zhang. "Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm." *Mathematical Programming Computation* 4.4 (2012): 333-361.]

LMaFit Model

- LMaFit aims at finding a low-rank (rank up to r) approximation W to M s.t. $\frac{1}{2}\|\mathcal{P}_\Omega(W-M)\|_F^2$ is minimized.
- The model is

$$\min_{X,Y,Z} \ \frac{1}{2}\|XY-Z\|_F^2 \ \text{ s.t. } \ Z_{ij}=M_{ij}, \forall (i,j)\in\Omega$$

  where $X \in \mathbb{R}^{m\times r}$, $Y \in \mathbb{R}^{r\times n}$, $Z \in \mathbb{R}^{m\times n}$.
- It is solved by a tuned version of block-coordinate descent algorithm or a nonlinear SOR .

LMaFit
Jacobi Iteration

- The simplest iterative method for the $Ax = b$ problem is Jacobi iteration.
- For the 3×3 case,

$$x_1 = (b_1 - a_{12}x_2 - a_{13}x_3)/a_{11},$$
$$x_2 = (b_2 - a_{21}x_1 - a_{23}x_3)/a_{22},$$
$$x_3 = (b_3 - a_{31}x_1 - a_{32}x_2)/a_{33}.$$

- If we write

$$L_A = \begin{bmatrix} 0 & 0 & 0 \\ a_{21} & 0 & 0 \\ a_{31} & a_{32} & 0 \end{bmatrix}, D_A = \begin{bmatrix} a_{11} & 0 & 0 \\ 0 & a_{22} & 0 \\ 0 & 0 & a_{33} \end{bmatrix}, U_A = \begin{bmatrix} 0 & a_{12} & a_{13} \\ 0 & 0 & a_{23} \\ 0 & 0 & 0 \end{bmatrix}$$

Jacobi iteration can be written as $D_A x^{(k)} = -(L_A + U_A)x^{(k-1)} + b$.

It will convergent if $\left\| D_A^{-1}(L_A + U_A) \right\| < 1$.

LMaFit
Gauss-Seidel Iteration

- If utilize the most recent solution estimate, we obtain the Gauss-Seidel iteration

$$\textbf{for } i = 1{:}n$$

$$x_i^{(k)} = \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)} \right) \Big/ a_{ii}$$

$$\textbf{end}$$

- It can be written as $(D_A + L_A)x^{(k)} = -U_A x^{(k-1)} + b$.
- The convergence rate is determined by $\|(D_A + L_A)^{-1} U_A\|$.

Fixed Point and Convergence

- Write

$$f(x) = -(D_A + L_A)^{-1} U_A x + (D_A + L_A)^{-1} b$$

  Then, the solution x^* is the fixed point of $f(x)$. i.e. $f(x^*) = x^*$.

- So,

$$f\left(x^{(k)} - x^*\right) = -(D_A + L_A)^{-1} U_A \left(x^{(k-1)} - x^*\right)$$

- We see that

$$\left\|x^{(k)} - x^*\right\| \leq \|(D_A + L_A)^{-1} U_A\| \left\|x^{(k-1)} - x^*\right\|$$

- If $\|(D_A + L_A)^{-1} U_A\| < 1$ the iteration $x^{(k)} = f\left(x^{(k-1)}\right)$ will converge to x^*.

LMaFit
SOR

- We can accelerate Gauss-Seidel by a slight modification by ω

$$(\omega D_A + L_A)x^{(k)} = -\big((1-\omega)D_A + U_A\big)x^{(k-1)} + b$$

- The idea is to choose ω s.t. $\left\|(\omega D_A + L_A)^{-1}\big((1-\omega)D_A + U_A\big)\right\|$ be small.
- This defines the method of **Successive Over-Relaxation (SOR).**

- LMaFit is a nonlinear SOR which modifies a nonlinear (block) Gauss-Seidel scheme.

LMaFit
Nonlinear Gauss-Seidel Method

- It is a straightforward alternating minimization scheme which updates the three variables X, Y, Z w.r.t. each one separately while fixing the other two.
- For example, by fixing Y and Z, compute the new X_+

$$X_+ = ZY^\dagger = \underset{X \in \mathbb{R}^{m \times K}}{\operatorname{argmin}} \ \frac{1}{2}\|XY - Z\|_F^2$$

  where $Y^\dagger$ is the *Moore-Penrose pseudo-inverse* of Y.
- The procedure is

$$\begin{aligned} X_+ &\leftarrow ZY^\dagger \\ Y_+ &\leftarrow (X_+)^\dagger Z \\ Z_+ &\leftarrow X_+Y_+ + \mathcal{P}_\Omega(M - X_+Y_+) \end{aligned}$$

LMaFit
Moore-Penrose Pseudo-Inverse

- We start by considering the minimization problem

$$x_b^* = \arg\min_{x \in \mathbb{R}^n} \|Ax - b\|_F^2$$

  where $A \in \mathbb{R}^{m \times n}$.
- If A has full column rank, this is the ordinary least square problem which has a geometric interpretation:
- Let $\mathrm{ran}(A)$ denote the space span by column vectors $A_1, A_2, \cdots, A_n$ of A. Then we have $Ax_b^* = b_\parallel$ where $b_\parallel$ is the orthogonal projection of $b = b_\perp + b_\parallel \in \mathbb{R}^m$ into $\mathrm{ran}(A)$.
- x_b^* is the coordinate of $b_\parallel$ under the basis of $A_1, A_2, \cdots, A_n$.

LMaFit
Moore-Penrose Pseudo-Inverse

- In the case that A does not have full column rank, the solution is not unique.
- All the solutions of $\arg\min_{x\in\mathbb{R}^n}\|Ax-b\|_F^2$ form a affine vector subspace

$$S_b = \{y \in \mathbb{R}^n : Ay = b_{\|}\}$$

  Let x_b^* be the shortest one in S_b.
- S_b is the translation of the vector space

$$S \coloneqq \{y \in \mathbb{R}^n : Ay = 0\} = \left(\mathrm{ran}(A^T)\right)^{\perp}$$

- x_* is the unique element in S_b that satisfies $x_* \perp S$, i.e. $x_* \in \mathrm{ran}(A^T)$.
  So x_* can be obtained by orthogonal projecting any element $y \in S_b$ to the subspace $\mathrm{ran}(A^T)$.

LMaFit
Moore-Penrose Pseudo-Inverse

- So, x_b^* for the problem $x_b^* = \arg\min_{x \in \mathbb{R}^n} \|Ax - b\|_F^2$ can be obtained by
- (1) orthogonally project b into $\mathrm{ran}(A)$ and get $b_{\|} \in \mathrm{ran}(A)$.
- (2) solve $Ay = b_{\|}$ and get any solution y.
- (3) orthogonally project y to $\mathrm{ran}(A^T)$ to get the shortest solution x_b^*.

- Since all the relations are linear, we can see

$$x^*_{(\lambda b_1 + \mu b_2)} = \lambda x^*_{b_1} + \mu x^*_{b_2}$$

LMaFit
Moore-Penrose Pseudo-Inverse

- Let x_i^* be the solution $x_{e_i}^*$ where $e_i = (0, \cdots, 0, 1, 0, \cdots, 0)^T \in \mathbb{R}^m$ and let

$$A^\dagger \coloneqq [x_1^* \, x_2^* \, \cdots \, x_m^*]$$

  be the $n \times m$ matrix composed of x_i^*.

- Then it's easy to see

$$x_b^* = A^\dagger b$$

- $A^\dagger$ is called the **Moore-Penrose Pseudo-Inverse** of A.

LMaFit
Moore-Penrose Pseudo-Inverse

- Consider the matrix version problem

$$X_B^* = \arg\min_{X \in \mathbb{R}^{n\times k}} \|AX - B\|_F^2$$

  where $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{m\times k}$ and $X \in \mathbb{R}^{n\times k}$.

- Let $B_1, B_2, \cdots, B_k$ be the column vectors of B and $X_1, X_2, \cdots, X_k$ be the column vectors of X.
- Observe that

$$(X_B^*)_i = \arg\min_{X_i \in \mathbb{R}^n} \|AX_i - B_i\|_F^2$$

- We have $X_B^* = A^\dagger B$.
- Similarly, $BA^\dagger = \arg\min_{X \in \mathbb{R}^{m\times n}} \|XA - B\|_F^2$ where $A \in \mathbb{R}^{n\times k}$, $B \in \mathbb{R}^{m\times k}$ and $X \in \mathbb{R}^{m\times n}$.

LMaFit
Moore-Penrose Pseudo-Inverse

- Pseudo-Inverse $A^{\dagger}$ can be computed by SVD of A.
- If $A \in \mathbb{R}^{n \times r}$ where $n \gg r$, we can use the relation

$$A^{\dagger} \equiv (A^T A)^{\dagger} A^T$$

  to accelerate computation.
- If A has full column rank, $A^{\dagger} = (A^T A)^{-1} A^T$.
- If A is invertible, then $A^{\dagger} = A^{-1}$.

LMaFit
Nonlinear SOR

- Using $\omega \geq 1$ as the extrapolation factor to the nonlinear Gauss-Seidel method, the nonlinear SOR method is

$$
\begin{aligned}
X_+ &\leftarrow ZY^\dagger \\
X_+(\omega) &\leftarrow \omega X_+ + (1-\omega)X \\
Y_+ &\leftarrow (X_+(\omega))^\dagger Z \\
Y_+(\omega) &\leftarrow \omega Y_+ + (1-\omega)Y \\
Z_+(\omega) &\leftarrow X_+(\omega)Y_+(\omega) + \mathcal{P}_\Omega(M - X_+(\omega)Y_+(\omega))
\end{aligned}
$$

- Basic Gauss-Seidel method correspond to $\omega = 1$.

LMaFit
Updating ω

- Define the residual ratio $\gamma(\omega)$

$$\gamma(\omega) = \frac{\left\|\mathcal{P}_\Omega\left(X^* - X_+(\omega)Y_+(\omega)\right)\right\|_F}{\|\mathcal{P}_\Omega(X^* - XY)\|_F}$$

- If $\gamma(\omega) < 1$, the step is "successful";
  Otherwise, the step is "unsuccessful". We reset $\omega = 1$ and try again.
- In the successful case, a small $\gamma(\omega)$ indicates a good choice of ω and we keep this value to the next iteration.
  If $\gamma(\omega)$ is not small enough, we increase ω for the next iteration hoping a better result.

LMaFit Algorism

1 Input index set Ω, data $\mathcal{P}_\Omega(M)$
2 Set $Y^0 \in \mathbb{R}^{r\times n}$, $Z^0 = \mathcal{P}_\Omega(M)$, $\omega = 1$, $\tilde{\omega} > 1$, $\delta > 0$, $\gamma_1 \in (0,1)$ and $k = 0$
3 **while** *not convergent* **do**
4 Compute $(X_+(\omega), Y_+(\omega), Z_+(\omega))$
5 Compute the residual ratio $\gamma(\omega)$
6 **if** $\gamma(\omega) \geq 1$ **then** set $\omega = 1$ and go to step 4
7 Update $(X^{k+1}, Y^{k+1}, Z^{k+1}) = (X_+(\omega), Y_+(\omega), Z_+(\omega))$ and increment k
8 **if** $\gamma(\omega) \geq \gamma_1$ **then** set $\delta = \max(\delta, 0.25(\omega - 1))$ and $\omega = \min(\omega + \delta, \tilde{\omega})$

LMaFit and Geometric Algorithms

- LMaFit has a geometric interpretation.
- Consider the simultaneous update version of LMaFit:

- $X_+ = ZY^\dagger = \big(XY + \mathcal{P}_\Omega(M - XY)\big)(Y^TY)^{-1}Y^T = X - \mathcal{P}_\Omega(XY - M)(Y^TY)^{-1}Y^T$
  $= X - S(Y^TY)^{-1}Y^T$
- $Y_+ = X^\dagger Z = (X^TX)^{-1}X^T\big(XY + \mathcal{P}_\Omega(M - XY)\big) = Y - (X^TX)^{-1}X^T\mathcal{P}_\Omega(XY - M)$
  $= Y - (X^TX)^{-1}X^TS$

  where $S \coloneqq \mathcal{P}_\Omega(XY - M)$.

- The SOR version is

- $X_+ = (1 - \omega)X + \omega(X - S(Y^TY)^{-1}Y^T) = X - \omega S(Y^TY)^{-1}Y^T$
- $Y_+ = (1 - \omega)Y + \omega(Y - (X^TX)^{-1}X^TS) = \mathrm{Y} - \omega(X^TX)^{-1}X^TS$

qGeomMC

- Consider MC problem

$$\arg\min_{X \in \mathbb{R}_r^{n\times m}} f(X) \coloneqq \frac{1}{2}\|\mathcal{P}_\Omega(X - X^*)\|_F^2$$

- Motivated by the factorization $X = GH^T$, where $G \in \mathbb{R}_*^{n\times r}$ and $H \in \mathbb{R}_*^{m\times r}$. qGeomMC optimize in the quotient space

$$M \coloneqq \mathbb{R}_r^{n\times m} = \mathbb{R}_*^{n\times r} \times \mathbb{R}_*^{m\times r}/GL(r)$$

where $GL(r)$ act on $\bar{M} \coloneqq \mathbb{R}_*^{n\times r} \times \mathbb{R}_*^{m\times r}$ as

$$GL(r) \times \bar{M} \to \bar{M}$$

$$\big(R, (G, H)\big) \mapsto (GR^{-1}, HR^T)$$

qGeomMC
Metric, Gradient

- The metric on the total space $\bar{M}$ is induced by a diagonal approximation of the Hessian of $\bar{f}(G,H) \coloneqq \frac{1}{2}\|\mathcal{P}_\Omega(GH - X^*)\|_F^2$.
- At $\bar{x} = (G,H) \in \bar{M}$, $T_{\bar{x}}\bar{M} = \mathbb{R}^{n\times r} \times \mathbb{R}^{m\times r}$, for $\bar{\xi}_{\bar{x}}, \bar{\eta}_{\bar{x}} \in T_{\bar{x}}\bar{M}$

$$\bar{g}_{\bar{x}}\left(\bar{\xi}_{\bar{x}}, \bar{\eta}_{\bar{x}}\right) \coloneqq tr\left((H^TH)\bar{\xi}_G^T\bar{\eta}_G\right) + tr\left((G^TG)\bar{\xi}_H^T\bar{\eta}_H\right)$$

- The gradient $\nabla\bar{f}$ at $\bar{x}$ under this metric is

$$\nabla\bar{f}(\bar{x}) = (SH(H^TH)^{-1}, S^TG(G^TG)^{-1})$$

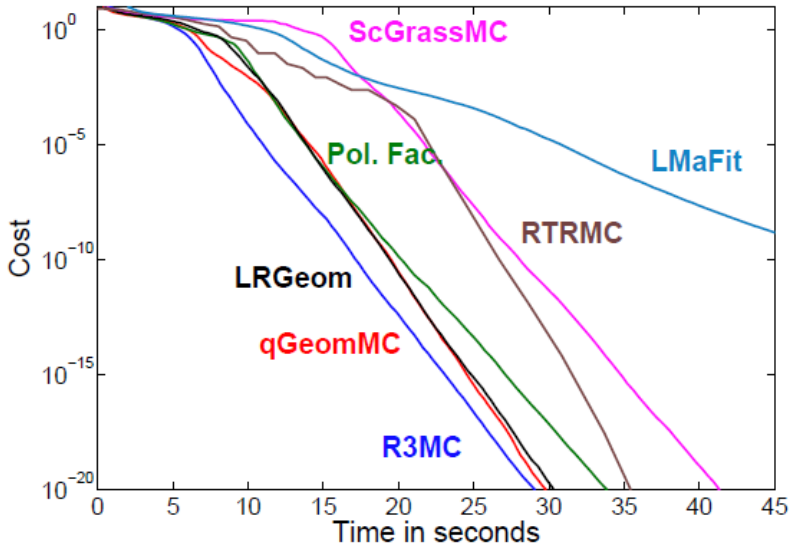
where $S \coloneqq \mathcal{P}_\Omega(GH^T - X^*)$.

qGeomMC
Gradient Descent Scheme

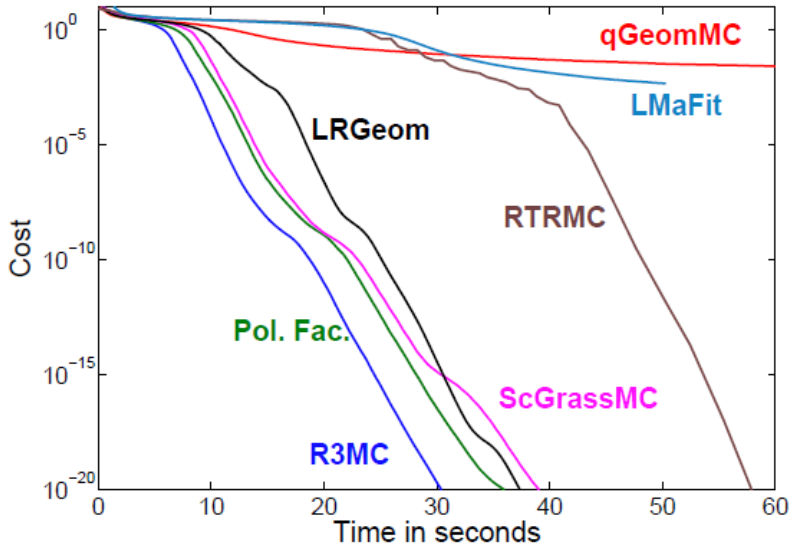
- The gradient descent's update rule is
- $G_+ = G - \omega SH(H^T H)^{-1}$
- $H_+ = H - \omega S^T G(G^T G)^{-1}$

- Compare this scheme with the simultaneous update version of LMaFit, we find that they share the same rule.

Comparison

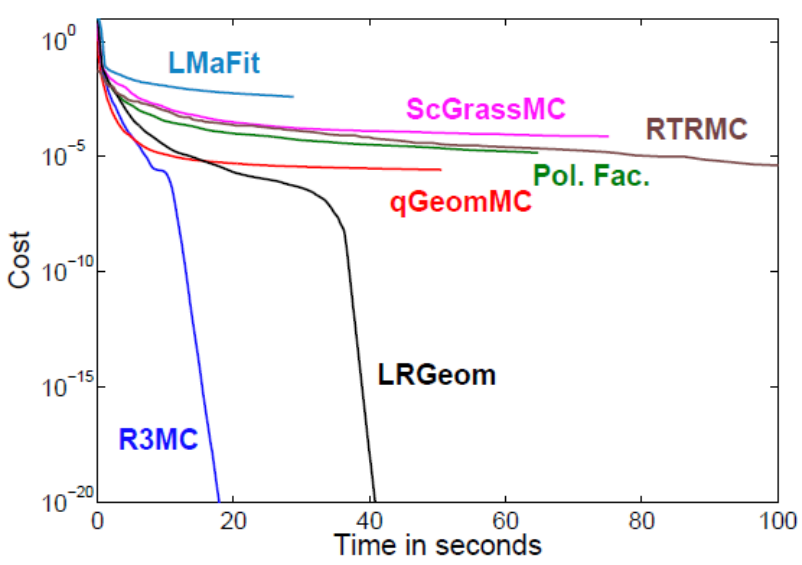


(a) OS = 3

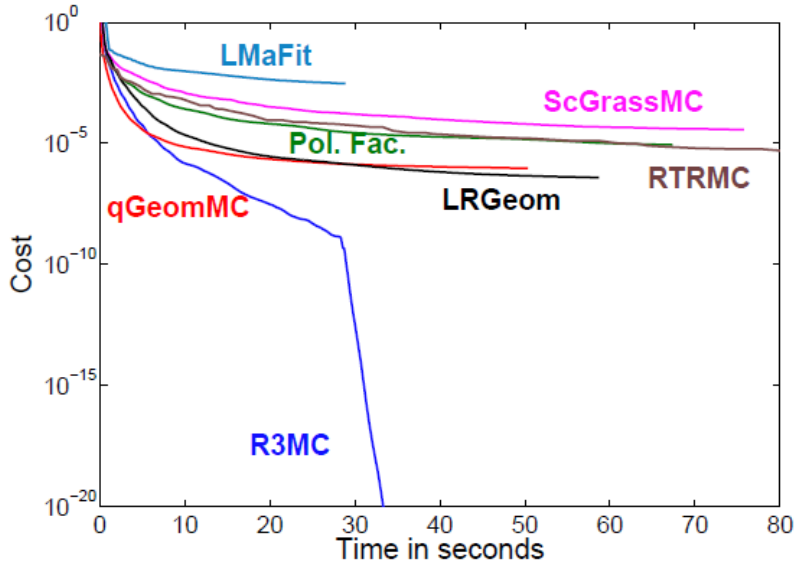


(b) OS = 2.5

Comparison



(a) CN = 100



(b) CN = 300

Storage

- All the algorithms do not compute directly on the $n \times m$ matrix but on the low-rank factorizations. This strategy save the required storage so that these algorithms can be applied to large scale datasets.

Computation

- All the algorisms need do SVD in each step. They use clever ways to avoid compute on the $n \times m$ matrix but on the low-rank parts. Which is workable.
- In general the computation cost is linear to $m + n$.

Conclusion

- We have introduced several optimization methods on the matrix manifolds which are efficient on MC problems.
- Optimization on manifolds have been considered long before, but the real application comes only recently partly due to the demand of the right problem to solve.

Thanks for Listening!

If you are interested in the topic, please feel free to stop by.