数理解析研究所講究録 第2108巻 2019年 168-176

リーマン多様体上の確率的最適化の発展

京都大学・大学院情報学研究科・数理工学専攻　佐藤 寛之

電気通信大学・大学院情報システム学研究科・情報ネットワークシステム学専攻　笠井 裕之

Microsoft, Bamdev Mishra

Hiroyuki Sato[1]

Department of Applied Mathematics and Physics, Graduate School of Informatics,

Kyoto University

Hiroyuki Kasai

Department of Information Network Systems, Graduate School of Information Systems,

The University of Electro-Communications

Bamdev Mishra

Microsoft

概 要

本稿では，幾何学的な最適化アルゴリズムの概要を解説するとともに，機械学習分野などにおいて現れる大規模問題に有効な確率的最適化手法について，リーマン多様体上への拡張に関する近年の研究を紹介する．特に，R-SRG とよばれる最新のアルゴリズムについて議論する．

1　はじめに

本稿では，連続最適化問題のうち，実行可能領域がリーマン多様体をなすような最適化問題を扱い，その求解アルゴリズムについて議論する．ユークリッド空間における制約なし問題に対する解法として，最急降下法や共役勾配法，ニュートン法などがよく知られているが，これらの手法は制約条件の情報を用いないため，そのままでは制約付き問題に適用することができない．しかし，実行可能領域がリーマン多様体 $\mathcal{M}$ であるような最適化問題は，$\mathcal{M}$ 上の無制約最適化問題と見なすことができる．そこで，ユークリッド空間における制約なし最適化問題に対する解法を多様体上に拡張する研究や，拡張されたアルゴリズムを他分野に応用する研究が盛んに行われている [1, 5, 12, 15]．

一方で，機械学習分野などにおけるビッグデータを伴う大規模な最適化問題を解く必要性から，ユークリッド空間における確率的最適化手法もやはり盛んに研究されている [6, 13, 14]．これらの手法では，反復計算によって最適化を行う際に，目的関数を構成する一部の項をランダムに選択し，それらの情報に基づいた解の更新を行う．すなわち，常に大規模なデータの全体を扱う代わりに，その一部をランダムに選ぶことで，現実的な計算時間で最適化を達成しようと試みるものである．

[1] e-mail: hsato[AT]amp.i.kyoto-u.ac.jp

以上の背景から，確率的最適化手法をリーマン多様体上に拡張する研究も近年注目を集めている．具体的には，Bonnabel [3] により，はじめて，リーマン多様体上での確率的勾配降下法（SGD）が提案されたが，近年の機械学習分野におけるユークリッド空間上でのSGDの拡張・高速化に呼応し，R-SVRG などの線形収束を達成する手法が提案されている [6]．さらに，リーマン多様体上での処理により適した R-SRG も提案された [8]．また，機械学習関係の主要な国際会議である NeurIPS (Neural Information Processing Systems), ICML (International Conference on Machine Learning), AISTATS (Artificial Intelligence and Statistics) などに採択された論文の中でも，リーマン多様体上の最適化を扱うものが多く見られる [4, 8, 9, 17, 18]．

本稿の構成は以下の通りである．2 節ではリーマン多様体上の最適化の一般論および，最適化において有用なレトラクションなどの幾何学的な概念を導入する．3 節ではリーマン多様体上の確率的最適化について，対象とする問題やいくつかの求解アルゴリズムを紹介するとともに，最新手法の一つとして R-SRG について議論する．最後に 4 節で本稿の結論を述べる．

2 リーマン多様体上の最適化について

2.1 リーマン多様体について

本稿で扱うリーマン多様体について簡単にまとめておく．まず，位相多様体を次のように定義する．

定義 2.1. 次の条件を満たす位相空間 $\mathcal{M}$ を n 次元位相多様体という．

1. $\mathcal{M}$ はハウスドルフ空間であり，第二加算公理を満たす．
2. $\mathcal{M}$ の任意の点 p に対して，p を含む $\mathcal{M}$ の開集合 U と，U から $\mathbb{R}^n$ のある開集合 V への同相写像 $\varphi\colon U \to V$ が存在する．

多様体 $\mathcal{M}$ 上の最適化のための反復アルゴリズムでは $\mathcal{M}$ 上の点列を生成するが，ハウスドルフ性によって，収束する点列の極限点の一意性が保証される．

また，本稿で考える多様体上の最適化では目的関数の微分の情報を用いる．多様体上の関数の微分について議論するためには，次のように定義される可微分多様体を扱う必要がある．

定義 2.2. r を正の整数または ∞ とする．$\mathcal{M}$ を n 次元位相多様体とし，$\{(U_\lambda, \varphi_\lambda)\}_{\lambda\in\Lambda}$ を $\mathcal{M}$ の座標近傍系とするとき，$U_\alpha \cap U_\beta \neq \emptyset$ なる任意の $\alpha, \beta \in \Lambda$ について $\varphi_\beta \circ \varphi_\alpha^{-1}\colon \varphi_\alpha(U_\alpha \cap U_\beta) \to \varphi_\beta(U_\alpha \cap U_\beta)$ が C^r 級写像であるならば，$\mathcal{M}$ を n 次元 C^r 級可微分多様体という．

以下では C^∞ 級可微分多様体を単に多様体ということにする．

さらに，多様体 $\mathcal{M}$ 上の最適化では，$\mathcal{M}$ 上の各点 x の接空間 $T_x\mathcal{M}$ において，接ベクトルの内積を扱う必要が生じる．そこで，各 $T_x\mathcal{M}$ に内積が与えられているような多様体，すなわち次のように定義されるリーマン多様体を考える．

定義 2.3. 多様体 $\mathcal{M}$ 上の 2 次の対称テンソル場 $\langle\cdot,\cdot\rangle\colon x \mapsto \langle\cdot,\cdot\rangle_x$（すなわち $\langle\cdot,\cdot\rangle_x$ は $T_x\mathcal{M}$ 上の双線形形式）が任意の点 $x \in \mathcal{M}$ において正定値であるとき，$\langle\cdot,\cdot\rangle$ を $\mathcal{M}$ 上のリーマン計量といい，リーマン計量を備えた多様体をリーマン多様体という．

定義 2.1 において多様体に第二可算性を課したことにより，多様体 $\mathcal{M}$ には少なくとも一つのリーマン計量が存在することが示される．

リーマン多様体 $\mathcal{M}$ においては滑らかな実数値関数 f の勾配 $\operatorname{grad} f$ を次のように定義することができ，$\mathcal{M}$ 上の目的関数の勾配の情報を用いることで，ユークリッド空間における勾配を用いたアルゴリズムを $\mathcal{M}$ 上に拡張することができる．

定義 2.4. リーマン多様体 $\mathcal{M}$ 上で定義された滑らかな実数値関数 f の点 $x \in \mathcal{M}$ における勾配 $\operatorname{grad} f(x)$ は，

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

が任意の接ベクトル $\xi \in T_x\mathcal{M}$ に対して成り立つような，$T_x\mathcal{M}$ の一意的なベクトルである．ここで，$\mathrm{D}f(x)\colon T_x\mathcal{M} \to T_{f(x)}\mathbb{R} \simeq \mathbb{R}$ は，f の x における微分である．

また，ヘッセ行列に対応する概念として，関数 f のヘシアン $\operatorname{Hess} f$ もリーマン計量を用いて定義することができる．目的関数のヘシアンはリーマン多様体上のニュートン法などで重要な役割を果たすが，本稿で紹介するアルゴリズム中では 1 階微分のみを用いるため，その詳細は割愛する．

2.2 最適化アルゴリズム

$\mathcal{M}$ をリーマン多様体とし，点 $x \in \mathcal{M}$ における接空間 $T_x\mathcal{M}$ の内積を $\langle\cdot,\cdot\rangle_x$ と書く．本節では，次のリーマン多様体 $\mathcal{M}$ 上の制約なし最適化問題を扱う．

問題 2.1.

$$\begin{aligned} &\text{minimize} \quad f(x), \\ &\text{subject to} \quad x \in \mathcal{M}. \end{aligned}$$

ユークリッド空間 $\mathbb{R}^n$ は，各点 $x \in \mathbb{R}^n$ における接空間 $T_x\mathbb{R}^n \simeq \mathbb{R}^n$ において標準内積が定まっていると見ることで，リーマン多様体の簡単な例であると見なすことができる．問題 2.1 において $\mathcal{M} = \mathbb{R}^n$ である場合は，目的関数 f の無制約最小化に対して直線探索法とよばれるアプローチがあり，初期点 $x_0 \in \mathbb{R}^n$ および

$$x_{k+1} = x_k + \alpha_k \eta_k, \quad k = 0, 1, 2, \ldots \tag{2.1}$$

によって点列 $\{x_k\}$ を生成する．ここで，$\alpha_k > 0$ はステップ幅，$\eta_k \in \mathbb{R}^n$ は探索方向とよばれる．探索方向は目的関数 f の勾配やヘッセ行列などの微分の情報を用いて計算されるベクトルである．

問題 2.1 における $\mathcal{M}$ がより一般のリーマン多様体である場合には，実行可能領域 $\mathcal{M}$ はベクトル空間であるとは限らないので，(2.1) の右辺の和が一般には定義されないため，

上述の直線探索をそのままの形で行うことはできない．そこで，点 $x_k \in \mathcal{M}$ を用いて次の点 $x_{k+1} \in \mathcal{M}$ を計算する際に，まず探索方向 η_k を x_k での接ベクトルとして選ぶ．さらに，x_k から $\eta_k \in T_{x_k}\mathcal{M}$ の方向に伸びる曲線 γ_{x_k,η_k} を考える．より正確には，曲線 γ_{x_k,η_k} に対して条件

$$\gamma_{x_k,\eta_k}(0) = x_k, \quad \dot{\gamma}_{x_k,\eta_k}(0) = \eta_k \tag{2.2}$$

を課す．この曲線 $\gamma_{x_k,\eta_k}(t)$ 上の $t = \alpha_k$ での f の値 $f(\gamma_{x_k,\eta_k}(\alpha_k))$ が十分減少するようにステップ幅 $\alpha_k > 0$ を求め，その α_k を用いて次の点 x_{k+1} を $x_{k+1} = \gamma_{x_k,\eta_k}(\alpha_k)$ により計算する．このような曲線を各反復において定義するには，以下で定義するレトラクションとよばれる写像を事前に用意しておくと便利である．ここで，レトラクション R は，点 x および x での接ベクトル η の対 (x, η) を引数とし，多様体 M 上の 1 点を与える写像，すなわち接バンドル $T\mathcal{M}$ から $\mathcal{M}$ への写像である．

定義 2.5. 次の性質を満たす写像 $R : T\mathcal{M} \to \mathcal{M}$ を $\mathcal{M}$ 上のレトラクションという．ただし，$R(x,\eta)$ を $R_x(\eta)$ と書くことにし，点 $x \in \mathcal{M}$ における接空間 $T_x\mathcal{M}$ の零ベクトルを 0_x とする．

1. $R_x(0_x) = x, \quad x \in \mathcal{M}.$
2. $\mathrm{D}R_x(0_x)[\eta] = \eta, \quad \eta \in T_x\mathcal{M}, \quad x \in \mathcal{M}.$

レトラクションの定義から，$\gamma_{x_k,\eta_k}(t) := R_{x_k}(t\eta_k)$ によって曲線 γ_{x_k,η_k} を定めれば，γ_{x_k,η_k} は性質 (2.2) を満たす曲線となる．したがって，x_{k+1} を計算するための更新式は

$$x_{k+1} = R_{x_k}(\alpha_k \eta_k) \tag{2.3}$$

となる．

更新式 (2.3) における探索方向を $\eta_k = -\operatorname{grad} f(x_k)$ として反復を行う最急降下法は簡明な最適化アルゴリズムではあるが，その収束は遅いことが知られており，共役勾配法やニュートン法など，多くの改良アルゴリズムがリーマン多様体においても研究されている．

次節では，特に大規模問題に着目し，確率的最適化手法を紹介する．その際，異なる接空間に属するベクトル同士の和に相当するものを計算する必要が生じる．それを可能にするのが次の vector transport とよばれる写像である．

定義 2.6. 次の性質を満たす写像 $\mathcal{T} : \bigcup_{x \in \mathcal{M}} (T_x\mathcal{M} \times T_x\mathcal{M}) \to T\mathcal{M}$ を $\mathcal{M}$ 上の vector transport という．ただし，$\xi, \eta \in T_x\mathcal{M}$ に対して，$\mathcal{T}(\xi, \eta)$ のことを $\mathcal{T}_\xi(\eta)$ と書くことにする．

1. レトラクション R が存在して，$\mathcal{T}_\xi(\eta) \in T_{R_x(\xi)}\mathcal{M}, \xi, \eta \in T_x\mathcal{M}, x \in \mathcal{M}.$
2. $\mathcal{T}_{0_x}(\eta) = \eta, \eta \in T_x\mathcal{M}, x \in \mathcal{M}.$
3. $\mathcal{T}_\xi(a\eta + b\zeta) = a\mathcal{T}_\xi(\eta) + b\mathcal{T}_\xi(\zeta), a, b \in \mathbb{R}, \xi, \eta, \zeta \in T_x\mathcal{M}, x \in \mathcal{M}.$

すなわち，vector transport はある接ベクトルを別の接空間に写す写像である．

3　リーマン多様体上の確率的最適化

3.1　問題設定といくつかのアルゴリズム

前節に引き続きリーマン多様体 $\mathcal{M}$ 上で定義された目的関数 f の最小化問題 2.1 を考えるが，本節では，f が

$$f(x) = \frac{1}{N}\sum_{n=1}^{N} f_n(x) \tag{3.1}$$

の形で表されている場合を議論する．ここで，自然数 N は非常に大きく，与えられた点 $x \in \mathcal{M}$ に対して目的関数値 $f(x)$ や勾配の値

$$\operatorname{grad} f(x) = \frac{1}{N}\sum_{n=1}^{N} \operatorname{grad} f_n(x)$$

を計算する際のコストが大きいという状況を想定する．具体的な応用例として，グラスマン多様体上の部分空間追跡法 [2] や，低ランクの行列補完問題およびテンソル補完問題 [7,11]，一定ランクの行列全体がなす多様体上の線形回帰問題 [10] などが，リーマン多様体上で定義された (3.1) の形の目的関数の最小化問題として定式化される．

以降では，このような設定の下で問題 2.1 を効率的に解くための方法を議論する．1 反復あたりの計算量を小さくするための簡単な方法の一つは，f の勾配 $\operatorname{grad} f$ の代わりに，ランダムに選んだ番号 $n \in \{1, 2, \ldots, N\}$ に対して f_n の勾配 $\operatorname{grad} f_n$ だけを用いて更新を行うことである．なお，以降では，計算コストの大きい $\operatorname{grad} f$ をフル勾配，ランダムな番号 n についての $\operatorname{grad} f_n$ を確率的勾配とよぶことにする．上記の考え方に基づくアルゴリズムを確率的勾配降下法（SGD）といい，$\mathcal{M}$ がユークリッド空間，一般のリーマン多様体である場合のどちらにも適用することができる．リーマン多様体の場合の SGD (R-SGD) は [3] により最初に提案され，同時に詳細な収束性解析も与えられた．(R-)SGD における更新を正確に述べると，$1, 2, \ldots, N$ のいずれかの値をランダムにとる確率変数 i_k を用いて

$$x_{k+1} = R_{x_k}\left(-\alpha_k \operatorname{grad} f_{i_k}(x_k)\right)$$

という計算を反復することになる．機械学習の文脈では，ステップ幅 α_k は学習率ともよばれる．ここで，$i_k = n$ となる確率が各 $n \in \{1, 2, \ldots, N\}$ について一様に $1/N$ ならば，$\operatorname{grad} f_{i_k}(x_k)$ の i_k についての期待値は

$$\mathbb{E}[\operatorname{grad} f_{i_k}(x_k)] = \frac{1}{N}\sum_{n=1}^{N} \operatorname{grad} f_n(x_k) = \operatorname{grad} f(x_k)$$

となり，$\operatorname{grad} f(x_k)$ と一致するから，確率的勾配 $\operatorname{grad} f_{i_k}(x_k)$ はフル勾配 $\operatorname{grad} f(x_k)$ の不偏推定量であり，確率的近似であるといえる．

このように (R-)SGD では，フル勾配 $\operatorname{grad} f$ を計算する場合に比べて，1 反復の計算量を大幅に小さくしているが，これは各反復においてごく一部のデータのみに注目して最適化を行っていることになるから，その収束は速くない．この問題点に対して，SGD の収束速度を改善するための一つの方法として，[6] によってユークリッド空間上の確率的

分散縮小勾配法（SVRG）が提案された．SVRG では，確率的勾配 $\mathrm{grad}\, f_{i_k}(x_k)$ に，フル勾配 $\mathrm{grad}\, f$ の情報を用いた補正項を付け加えて分散が小さくなるように工夫する．ただし，フル勾配 $\mathrm{grad}\, f$ は各反復において計算するわけではなく，相当数の反復を行う度に 1 回だけ計算し，その計算結果を以降の反復でしばらく用い続ける．この補正により，確率的に定まる探索方向の分散を小さくするという効果が生まれる．さらに，SVRG をリーマン多様体上に拡張した [18] では，指数写像や平行移動を用いたアルゴリズムが提案されているが，実際の数値計算ではレトラクションや vector transport を用いることが望ましい．これを実現した [16] によるアルゴリズム R-SVRG を以下に示す．なお，R-SVRG の詳細は [19] でも解説されている．

アルゴリズム 1 R-SVRG [16]

Require: $m_s > 0$ と学習率の列 $\{\alpha_t^s\}$.

$\tilde{x}^0$ を初期化する．
for $s = 1, 2, \ldots$ **do**
  フル勾配 $\mathrm{grad}\, f(\tilde{x}^{s-1})$ を計算し，$x_0^s = \tilde{x}^{s-1}$ を保存する．
  for $t = 1, 2, \ldots, m_s$ **do**
    $i_t^s \in \{1, 2, \ldots, N\}$ を一様にランダムに選択する．
    $\tilde{x}^{s-1}$ から x_{t-1}^s へ向かう接ベクトル ζ を $\zeta = R_{\tilde{x}^{s-1}}^{-1}(x_{t-1}^s)$ により計算する．
    ξ_t^s を vector transport を用いて

$$\xi_t^s = \mathrm{grad}\, f_{i_t^s}(x_{t-1}^s) - \mathcal{T}_\zeta \left(\mathrm{grad}\, f_{i_t^s}(\tilde{x}^{s-1}) - \mathrm{grad}\, f(\tilde{x}^{s-1})\right)$$

    により計算する．
    レトラクション R を用いて x_t^s を $x_t^s = R_{x_{t-1}^s}\left(-\alpha_{t-1}^s \xi_t^s\right)$ と更新する．
  end for
  $\tilde{x}^s = x_{m_s}^s$.
end for

3.2 Riemannian stochastic recursive gradient algorithm

本項では，最新の確率的最適化手法の一つとして，stochastic recursive gradient algorithm (SRG) およびそれをリーマン多様体上に拡張した Riemannian SRG (R-SRG) を紹介する．ユークリッド空間の場合の SRG は [14] で提案されており，本節ではより一般の場合の R-SRG [8] を紹介する．R-SVRG (アルゴリズム 1) では各内部反復においてレトラクション R の逆 R^{-1} の計算が必要であるのに対し，R-SRG ではそのような計算が不要であるなど，SRG は SVRG よりもユークリッド空間からリーマン多様体上への一般化に適した手法である．

前項で紹介した R-SVRG と同様に，R-SRG も二重のループ構造をもつアルゴリズムである．しかし，R-SVRG とは異なり，R-SRG の内部反復では，第 t 反復における確率的勾配を修正したベクトル v_t を，一つ前の v_{t-1} から確率的勾配を足し引きすることによっ

て計算する．より具体的には，$v_0 = \mathrm{grad}\, f(x_0)$ とし，$t \geq 1$ に対して

$$v_t = \mathrm{grad}\, f_{i_t}(x_t) - \mathcal{T}_{x_{t-1}}^{x_t} \mathrm{grad}\, f_{i_t}(x_{t-1}) + \mathcal{T}_{x_{t-1}}^{x_t} v_{t-1} \tag{3.2}$$

とする．ここで，式 (3.2) の右辺の第 1 項は $T_{x_t}\mathcal{M}$ のベクトルであり，第 2 項，第 3 項に現れる $\mathrm{grad}\, f_{i_t}(x_{t-1})$ や v_{t-1} は $T_{x_{t-1}}\mathcal{M}$ のベクトルであるため，後者に vector transport $\mathcal{T}$ を作用させることで $T_{x_t}\mathcal{M}$ のベクトルに写し，ベクトルの加減の計算を実現していることに注意する．この v_t を用いて，最適化問題 2.1 の解の候補点 x_t をレトラクション R により

$$x_{t+1} = R_{x_t}(-\alpha_t v_t).$$

と更新する．なお，R-SVRG では，修正された確率的勾配がフル勾配の不偏推定量であったのに対して，R-SRG では一般には状況が異なる．すなわち，点 x_{t-1} や方向 v_{t-1} が与えられている下で，i_t についての v_t の平均は $\mathrm{grad}\, f(x_t) - \mathcal{T}_{x_{t-1}}^{x_t} \mathrm{grad}\, f(x_{t-1}) + \mathcal{T}_{x_{t-1}}^{x_t} v_{t-1} \neq \mathrm{grad}\, f(x_t)$ である．しかしながら，反復全体を通しての期待値は $\mathbb{E}[v_t] = \mathbb{E}[\mathrm{grad}\, f(x_t)]$ を満たす．R-SRG のアルゴリズムは以下の通りである．

アルゴリズム **2** R-SRG [8]

Require: $m > 0$ と学習率の列 $\{\alpha_t\}$.

$\tilde{x}^0$ を初期化する．

for $s = 1, 2, \ldots$ **do**

　$x_0 = \tilde{x}^{s-1}$ を保存する．

　フル勾配 $\mathrm{grad}\, f(x_0)$ を計算する．

　$v_0 = \mathrm{grad}\, f(x_0)$ を保存する．

　$x_1 = R_{x_0}(-\alpha_0 v_0)$ と更新する．

　for $t = 1, 2, \ldots, m-1$ **do**

　　$i_t \in \{1, 2, \ldots, N\}$ を一様にランダムに選択する．

　　v_t を vector transport を用いて

$$v_t = \mathrm{grad}\, f_{i_t}(x_t) - \mathcal{T}_{x_{t-1}}^{x_t} \mathrm{grad}\, f_{i_t}(x_{t-1}) + \mathcal{T}_{x_{t-1}}^{x_t} v_{t-1}$$

　　によって計算する．

　　$x_{t+1} = R_{x_t}(-\alpha_t v_t)$ と更新する．

　end for

　ランダムに選択した $t' \in \{0, 1, \ldots, m\}$ に対して $\tilde{x}^s = x_{t'}$ とする．

end for

R-SRG の収束性の議論についての詳細は [8] を参照されたい．ここでは， [8] で議論されている一連の結果の一つとして，f が $\mathcal{S} \subset \mathcal{M}$ において，レトラクション R についての retraction convex とよばれる性質を満たす場合，すなわち，任意の $x \in \mathcal{S}$ および $\|\eta\|_x = 1$ を満たす任意の $\eta \in T_x\mathcal{M}$ に対して，$f(R_x(\tau\eta))$ が $f(R_x(\tau\eta)) \in \mathcal{S}$ を満たす任意の τ に対して凸関数である場合の結果を紹介する．このとき，いくつかの仮定の下で，十分小さ

い一定値のステップ幅 $\alpha_t \equiv \alpha$ を用いたアルゴリズム 2 により生成される点列 $\{\tilde{x}^s\}$ に対して，ある $\Delta > 0$, $\phi \in (0,1)$ が存在して，

$$\mathbb{E}[\|\operatorname{grad} f(\tilde{x}^s)\|_{\tilde{x}^s}^2] - \Delta \leq \varphi^s(\|\operatorname{grad} f(\tilde{x}^0)\|_{\tilde{x}^0}^2 - \Delta)$$

が成り立つ [8]．ここで，アルゴリズム 2 における各内部反復の反復回数 m を十分大きく選べば Δ は十分小さくできる．

4　結論

本稿では，リーマン多様体上の最適化の一般論を説明した後，リーマン多様体上の大規模最適化問題に対して有効な確率的最適化を紹介した．特に，最新の研究成果である R-SRG について，そのアルゴリズムと収束性を紹介した．確率的最適化アルゴリズムはユークリッド空間においても盛んに研究が続けられており，最先端の結果も引き続きリーマン多様体へと拡張されることが期待される．

謝辞

本研究は JSPS 科研費 JP16K17647, JP16K00031, JP17H01732 の助成を受けている．

参考文献

[1] P.-A. Absil, R. Mahony and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*, Princeton University Press, 2008.

[2] L. Balzano, R. Nowak and B. Recht, Online identification and tracking of subspaces from highly incomplete information, *Proceedings of the 48th Annual Allerton Conference on Communication, Control, and Computing* (Allerton 2010), 704–711.

[3] S. Bonnabel, Stochastic gradient descent on Riemannian manifolds, *IEEE Transactions on Automatic Control*, **58**(9) (2013), 2217–2229.

[4] A. Douik and B. Hassibi, Low-rank Riemannian optimization on positive semidefinite stochastic matrices with applications to graph clustering, *Proceedings of Machine Learning Research*, **80** (ICML 2018), 1299–1308.

[5] A. Edelman, T. A. Arias and S. T. Smith, The geometry of algorithms with orthogonality constraints, *SIAM Journal on Matrix Analysis and Applications*, **20** (1998), 303–353.

[6] R. Johnson and T. Zhang, Accelerating stochastic gradient descent using predictive variance reduction, *Advances in Neural Information Processing Systems*, **26** (NeurIPS 2013), 315–323.

[7] H. Kasai and B. Mishra, Low-rank tensor completion: a Riemannian manifold preconditioning approach, *Proceedings of Machine Learning Research*, **48** (ICML 2016), 1012–1021.

[8] H. Kasai, H. Sato and B. Mishra, Riemannian stochastic recursive gradient algorithm, *Proceedings of Machine Learning Research*, **80** (ICML 2018), 2516–2524.

[9] H. Kasai, H. Sato and B. Mishra, Riemannian stochastic quasi-Newton algorithm with variance reduction and its convergence analysis, *Proceedings of Machine Learning Research*, **84** (AISTATS 2018), 269–278.

[10] G. Meyer, S. Bonnabel and R. Sepulchre, Linear regression under fixed-rank constraints: a Riemannian approach, *Proceedings of the 28th International Conference on Machine Learning* (ICML 2011).

[11] B. Mishra and R. Sepulchre, R3MC: A Riemannian three- factor algorithm for low-rank matrix completion, *Proceedings of the 53rd IEEE Conference on Decision and Control* (CDC 2014), 1137–1142.

[12] B. Mishra and R. Sepulchre, Riemannian preconditioning, *SIAM Journal on Optimization*, **26** (2016), 635–660.

[13] P. Moritz, R. Nishihara and M. I. Jordan, A linearly-convergent stochastic L-BFGS algorithm, *Proceedings of Machine Learning Research*, **51** (AISTATS 2016), 249–258.

[14] L. M. Nguyen, J. Liu, K. Scheinberg and M. Takáč, SARAH: A novel method for machine learning problems using stochastic recursive gradient, *Proceedings of Machine Learning Research*, **70** (ICML 2017), 2613–2621.

[15] H. Sato, A Dai–Yuan-type Riemannian conjugate gradient method with the weak Wolfe conditions, *Computational Optimization and Applications*, **64** (2016), 101–118.

[16] H. Sato, H. Kasai and B. Mishra, Riemannian stochastic variance reduced gradient, arXiv preprint, arXiv:1702.05594 (2017).

[17] Z. Xu and X. Gao, On truly block eigensolvers via Riemannian optimization, *Proceedings of Machine Learning Research*, **84** (AISTATS 2018), 168–177.

[18] H. Zhang, S. J. Reddi and S. Sra, Riemannian SVRG: Fast stochastic optimization on Riemannian manifolds, *Advances in Neural Information Processing Systems*, **29** (NeurIPS 2016), 4592–4600.

[19] 佐藤寛之，笠井裕之，Bamdev Mishra，直交制約つき最適化問題に対するリーマン多様体上の確率的分散縮小勾配法，京都大学数理解析研究所講究録，**2027** (2017), 135–143.