Escaping saddle points efficiently in equality-constrained optimization problems

Yue Sun [1] **Maryam Fazel** [1]

Abstract

We consider minimizing a nonconvex, smooth function $f(x)$ subject to equality constraints $c_i(x) = 0$ (equivalently, $x \in \mathcal{M}$ where $\mathcal{M}$ is a smooth manifold). We show that a perturbed version of the gradient projection algorithm converges to a second-order stationary point for this problem (and hence is able to escape saddle points on the manifold) in a number of iterations that depend only polylogarithmically on the dimension (hence is almost dimension-free). This matches a rate known only for unconstrained smooth minimization. While the unconstrained case is well-studied, our result is the first to prove such a rate for a constrained problem, which includes examples such as PCA, Burer-Monteiro factorized SDPs, and more. The rate of convergence depends as $1/\epsilon^2$ on the accuracy ϵ, and also depends polynomially on appropriate smoothness and curvature parameters for the cost function and the constraints – we define these parameters using the explicit form of the constraints $c_i(x) = 0$ (in a representation-dependent fashion), but also briefly examine a geometric alternative for the manifold curvature. Future work will examine this geometric setting further, and will consider the more challenging problems of inequality constraints.

1. Introduction

We consider the optimization problem

$$\underset{x}{\text{minimize}} \;\; f(x), \quad \text{subject to } \; c_i(x) = 0, \; i = 1, ..., m, \tag{1}$$

where $x \in \mathbb{R}^d$ is the optimization variable, the constraint set $\{x \mid c_i(x) = 0\}$ defines a (smooth) manifold $\mathcal{M} \subset \mathbb{R}^d$, and the function $f(x)$ (nonconvex in general) is twice differentiable, with a Hessian that is ρ-Lipschitz [1].

First-order optimization methods such as gradient descent and its variants (e.g., stochastic gradients) and gradient projection methods (for constrained problems where projections can be computed) are widely used in machine learning applications due to their simplicity and favorable computational properties. When the problem is not convex, much of the literature focuses on rates of converges to first-order stationary points. However, an important practical consideration is how fast the algorithm converges to a local minimum and not just any stationary point—that is, whether the algorithm can *escape from a saddle point* efficiently, which also means it can avoid slowing down significantly when passing near a saddle point. This question has attracted much interest (Lee et al., 2016; Du et al., 2017; Ge et al., 2015; Jin et al., 2017a;b), leading to a set of results for the smooth unconstrained minimization, reviewed below.

Smooth unconstrained minimization. If $\mathcal{M}$ is the entire space $\mathbb{R}^d$, the problem reduces to an unconstrained nonconvex minimization problem. The algorithms as well as their convergence rates are well understood in the convex case. In the nonconvex case, existing analysis often focuses on the rate of convergence to a first order stationary point (where the gradient is 0), and in this case gradient descent is still provably powerful. To show convergence to second-order stationary points, the algorithm has to move away from saddle points. In an asymptotic sense, this is not an issue, since it is known that gradient descent starting from a random initial point does not converge to a saddle point (with probability one) (Pemantle, 1990; Lee et al., 2016). However, it is still important to quantify the rates: (Du et al., 2017)

[1]Department of Electrical Engineering, University of Washington, Seattle, United States. Correspondence to: Yue Sun <yuesun@uw.edu>, Maryam Fazel <mfazel@uw.edu>.

Workshop on Modern Trends in Nonconvex Optimization for Machine Learning. The 35^{th} International Conference on Machine Learning, Stockholm

[1]This problem can also be considered purely geometrically, independent of the representation of the manifold. In this paper, we focus on explicit constraints given by $c_i(x)$, which is more natural in many cases, and so that our analysis will reveal the effect of a specific representation on algorithm performance. However, we also mention connections to intrinsic manifold parameters such as the manifold curvature, and in the subsection 2.1 we comment on developing similar results in purely geometric terms.

shows that gradient descent can be exponentially slow in the presence of saddle points, and even passing near a saddle point can cause a significant slow down.

One way to address this issue is the cubic regularization algorithm proposed in (Nesterov & Polyak, 2016) Define a $(\epsilon, -\sqrt{\rho\epsilon})$ stationary point where x satisfies $\|\nabla f(x)\| \leq \epsilon$, $\lambda_{\min}(\nabla^2 f(x)) \geq -\sqrt{\rho\epsilon}$. When we have access to the Hessian or a Hessian-vector product oracle, the algorithm (Carmon & Duchi, 2017) returns a $(\epsilon, -\sqrt{\rho\epsilon})$ stationary point in $\mathrm{polylog}(d)$ iterations. This algorithm requires a Hessian vector product oracle, and when this oracle can be implemented efficiently (typically by taking the difference of the gradients at two close points, i.e., $Hv = \lim_{\epsilon\to 0} \frac{\nabla f(x+\epsilon v)-\nabla f(x)}{\epsilon}$), this algorithm is powerful. As a variation, trust-region algorithm is proposed in (Sun et al., 2015; 2016) and has the same convergence guarantee.

Another line of work is based on simply adding noise to the gradient to escape from a saddle point. Although gradient descent (without noise) can be exponentially slow in escaping from saddle points (Du et al., 2017), for noisy gradient descent, (Ge et al., 2015) and (Jin et al., 2017a) prove the convergence of stochastic and perturbed gradient descent to an approximate local minimizer. Compared to Hessian vector product, gradient descent is simple to apply and is used broadly in machine learning practice. (Jin et al., 2017b) analyzes the noise-perturbed heavy ball method, which increases convergence rate from $\tilde{O}(\epsilon^{-2})$ to $\tilde{O}(\epsilon^{-7/4})$. (Allen-Zhu, 2017) proposes a stochastic gradient method with a reasonable convergence rate $\tilde{O}(\epsilon^{-3.25})$ to converge to $(\epsilon, -\epsilon^{1/4})$ stationary points.

Equality constrained minimization. For a problem with equality constraints (or a manifold constraint) the convergence to first order stationary points is well studied (e.g., (Tripuraneni et al., 2018; Zhang et al., 2016)). Also (Boumal et al., 2018) shows that second order algorithms, such as trust region methods, converge to second order stationary points.

With exception of appendix B of (Ge et al., 2015), all existing results on rates of escape from saddles with (noisy) first order methods apply only to smooth unconstrained problems. In this case, the Hessian of the function captures the local curvature, and the direction that decreases the function value is a negative eigenvector of the Hessian.

Far less is known about constrained problems; even those with equality (or manifold) constraints only. The only known rate analysis is given in (Ge et al., 2015) for stochastic projected gradient descent on a manifold parameterized by a set of equalities $c_i(x) = 0$, $i = 1, ..., m$ where each constraint $c_i(x)$ is smooth. The convergence rate, however, is *polynomial* in the problem dimension d, mainly due to the fact that "too much noise" is introduced in the stochastic oracle.

Contributions. In this paper, we show that for the constrained optimization problem of minimizing $f(x)$ subject to a manifold constraint (or a set of equality constraints $c_i(x) = 0$), as long as the function and the manifold are appropriately smooth, a perturbed projected gradient descent algorithm will escape saddle points with a rate that has an $1/\epsilon^2$ dependence on the accuracy ϵ, a polynomial dependence on the curvature and smoothness parameters, and more importantly, a *polylogarithmic* dependence on the problem dimension (hence the complexity is almost dimension-free). This improves the rate of the best known result for equality-constrained optimization, which was polynomial in dimension(Ge et al., 2015). We also give the explicit dependence of the rate of convergence on the smoothness parameters.

2. Perturbed projected gradient descent with equality constraints

We begin by defining parameters that capture the smoothness and curvature of the manifold in problem (1). Using the notation in (Ge et al., 2015), we represent the manifold by equalities $c_i(x) = 0, i = 1, ..., m$, and assume $c_i(x)$ has β_i-smooth gradients. We also assume constraints $c_i, i = 1, ..., m$ satisfy the so-called α_c-RLICQ (robust linear independence constraint qualification) (Ge et al., 2015), i.e., the smallest singular value of the matrix $C = [\nabla c_1(x), ..., \nabla c_m(x)]$ is α_c. This is a robust variant of the well known LICQ (Wright & Nocedal, 1999).

Define the parameter $R = (\sum_{i=1}^{m} \frac{\beta_i^2}{\alpha_c^2})^{-1/2} = \alpha_c/\sqrt{\sum_i \beta_i^2}$, which can be seen as a lower bound on the radius of the manifold. Lemma 1 is a restatement of the geometric lemmas 25-27 in (Ge et al., 2015).

Lemma 1. *(Ge et al., 2015) We have the following conclusions about R,*

- $\forall x, x_0 \in \mathcal{M}$, $\|\mathcal{P}_{\mathcal{T}^c_{x_0}}(x - x_0)\| \leq \frac{1}{2R}\|x - x_0\|^2$,
- $\forall x, x_0 \in \mathcal{M}$, $y \in \mathcal{T}^c_x$, $\|\mathcal{P}_{\mathcal{T}_{x_0}} y\| \leq \frac{1}{R}\|x - x_0\|\|y\|$,
- $\forall x \in \mathcal{M}$, $y \in \mathcal{T}_x$, $\|x + y - \mathcal{P}_{\mathcal{M}}(x + y)\| \leq \frac{4\|y\|^2}{R}$.

Here, $\mathcal{T}_x$ denotes the tangent space of $\mathcal{M}$ at x, $\mathcal{T}^c_x$ is its orthogonal complement (the normal space), and $\mathcal{P}$ denotes orthogonal projection onto a subspace. The following is our main theorem, which gives the convergence rate of Algorithm 1 to a second-order stationary point (therefore escaping saddles) in time $1/\epsilon^2$ and with only polylogarithmic dependence on the dimension d. This is the first almost dimension-free rate for a nonconvex constrained problem using (noisy) first order methods.

The parameters appearing here are all associated with the smoothness of the Lagrangian $L(x, \lambda^*(x))$: l denotes its Lipshitz constant of $L(x, \lambda^*)$, β_L is the Lipshitz constant of its gradient, and ρ is the Lipshitz constant of its Hessian.

Theorem 1. *Let $L(x, \lambda^*)$ denote the Lagrangian for problem (1) with $\lambda^*(x) = \operatorname{argmin}_\lambda \|\nabla_x L(x, \lambda)\|_2$. Suppose $L(x, \lambda^*)$ is l Lipschitz, β_L smooth, has a ρ Lipschitz Hessian, and the constraint manifold has radius R. Then with probability $1 - \delta$, Algorithm 1 takes $O(\max\{1, (\frac{l}{\beta_L R})^2, (\frac{\beta_L}{\rho R})^3\} \frac{\beta_L (f(x_0) - f^*)}{\epsilon^2} \log^4(\frac{d\beta_L (f(x_0) - f^*)}{\epsilon^2 \delta}))$ iterations to reach an $(\epsilon, -\sqrt{\rho\epsilon})$-stationary point.*

2.1. Relation to geometric manifold parameters

In this subsection, we examine how the parameter R, which we defined based on the representation given by c_i, relates to the intrinsic (representation independent) manifold curvature. Let $\operatorname{dist}(x) : \mathbb{R}^d \to \mathbb{R}$ denote the distance to $\mathcal{M}$,

$$\operatorname{dist}(x) = \min_{u \in \mathcal{M}} \|x - u\|.$$

The tangent space and normal space are defined by directional derivative of $\operatorname{dist}(x)$, i.e., $\operatorname{dist}'(x; v) = \|\mathcal{P}_{\mathcal{T}_x^c}(v)\|, x \in \mathcal{M}$. For a manifold, define the smoothness parameter $\beta_{\mathcal{M}}$ by

$$\left|\operatorname{dist}(x + y) - \operatorname{dist}(x) - \|\mathcal{P}_{\mathcal{T}_x^c}(y)\|\right| \le \frac{1}{2}\beta_{\mathcal{M}}\|y\|^2, x \in \mathcal{M}. \tag{2}$$

Note that $x \in \mathcal{M}$, so $\operatorname{dist}(x) = 0$. Now we restate Lemma 1 in terms of $\beta_{\mathcal{M}}$ which does not depend on the representation. We show in the supplement that the three inequalities in Lemma1 hold if we set $R = 8/\beta_{\mathcal{M}}$. It follows that Theorem 1 can also be expressed in terms of parameter $\beta_{\mathcal{M}}$.

Theorem 2. *Suppose $f(x)$ is smooth in $\mathbb{R}^d$, and the optimization problem is defined on a manifold with smoothness parameter $\beta_{\mathcal{M}}$. Suppose $f(x)$ is smooth on the manifold with the following parameters $\|\mathcal{P}_{\mathcal{T}_x} \nabla f(x)\| \le l$, $\|\nabla_x \mathcal{P}_{\mathcal{T}_x}(\nabla f(x))\| \le \beta_L$, and $\|\nabla_x \mathcal{P}_{\mathcal{T}_x}(\nabla f(x)) - \nabla_y \mathcal{P}_{\mathcal{T}_y}(\nabla f(y))\| \le \rho\|x - y\|$. Then with probability $1 - \delta$, Algorithm 1 converges to a $(\epsilon, -\sqrt{\rho\epsilon})$-stationary point (satisfying $\|\mathcal{P}_{\mathcal{T}_x} \nabla f(x)\| \le \epsilon$, $v^T \nabla_x \mathcal{P}_{\mathcal{T}_x}(\nabla f(x)) v \le -\sqrt{\rho\epsilon}$ for all $v \in \mathcal{T}_x$, $\|v\| = 1$) in*

$$O(\max\{1, (\frac{l\beta_{\mathcal{M}}}{\beta_L})^2, (\frac{\beta_L \beta_{\mathcal{M}}}{\rho})^3\} \cdot \frac{\beta_L (f(x_0) - f^*)}{\epsilon^2} \log^4(\frac{d\beta_L (f(x_0) - f^*)}{\epsilon^2 \delta}))$$

iterations.

3. Proof Sketch

The proof adapts ideas from (Jin et al., 2017a), which is restricted to the unconstrained problem. For ease of notation, we let $R = 8/\beta_{\mathcal{M}}$ in the proof, and use Lemma 1.

Algorithm 1 Perturbed projected gradient algorithm

Require: Function $f(x)$, initial point $x_0 \in \mathbb{R}^d$, parameters l, β_L, ρ (associated with the Lagrangian), accuracy ϵ. $\mathcal{M}$ is the manifold defined by $c_i(x) = 0$, $i \in [m]$, with projection $\mathcal{P}_{\mathcal{M}}$; $\mathcal{T}_x$ is the tangent space to $\mathcal{M}$ at x. $\mathbb{B}_0(r)$ is the Euclidean ball of radius r.
Set constants: $\nu = \frac{\rho R}{\rho R + 8\beta_L}$, $r = \frac{c\nu}{\chi^2}\epsilon$, $\chi = 3\max\{\log(\frac{d\beta_L \Delta_f}{c\epsilon^2\delta}), 4\}$.
Set threshold values: $f_{\text{thres}} = \frac{c\nu}{\chi^3}\sqrt{\frac{\epsilon^3}{\rho}}$, $g_{\text{thres}} = \frac{\sqrt{c\nu}}{\chi^2}\epsilon$, $t_{\text{thres}} = \frac{\chi}{c^2}\frac{\beta_L}{\sqrt{\rho\epsilon}}$, $t_{\text{noise}} = -t_{\text{thres}} - 1$.
Set stepsize: $\eta = \frac{c_{\max}}{\beta_L}$, $c_{\max} = O(1, \beta_L R/l, \rho R/\beta_L)$ chosen properly (described in appendix).
while 1 **do**
  if $\|\mathcal{P}_{\mathcal{T}_{x_t}} \nabla f(x_t)\| \le g_{\text{thres}}$ and $t - t_{\text{noise}} > t_{\text{thres}}$ **then**
    $t_{\text{noise}} \leftarrow t$, $\tilde{x}_t \leftarrow x_t$, $x_t \leftarrow \mathcal{P}_{\mathcal{M}}(x_t + \xi_t)$, ξ_t uniformly sampled from $\mathbb{B}_0(r)$.
  end if
  if $t - t_{\text{noise}} = t_{\text{thres}}$ and $f(x_t) - f(\tilde{x}_{t_{\text{noise}}}) > -f_{\text{thres}}$ **then**
    output $\tilde{x}_{t_{\text{noise}}}$
  end if
  $t \leftarrow t + 1$.
  $x_t \leftarrow \mathcal{P}_{\mathcal{M}}(x_{t-1} - \eta \mathcal{P}_{\mathcal{T}_{x_{t-1}}} \nabla f(x_{t-1}))$.
end while

3.1. Case 1: When the gradient is large

We first consider the case where the norm of the gradient is large, i.e., $\|\mathcal{P}_{\mathcal{T}_x} \nabla f(x)\| \ge \varepsilon$. Then

$$f(x - \eta \mathcal{P}_{\mathcal{T}_x} \nabla f(x)) \le f(x) - \eta \|\mathcal{P}_{\mathcal{T}_x} \nabla f(x)\|^2 + \frac{\eta^2 \beta_L}{2} \|\mathcal{P}_{\mathcal{T}_x} \nabla f(x)\|^2.$$

And the increase in the function value caused by projection is upper bounded by

$$\begin{aligned} & f(\mathcal{P}_{\mathcal{M}}(x - \eta \mathcal{P}_{\mathcal{T}_x} \nabla f(x))) - f(x - \eta \mathcal{P}_{\mathcal{T}_x} \nabla f(x)) \\ \le\ & l \|\mathcal{P}_{\mathcal{M}}(x - \eta \mathcal{P}_{\mathcal{T}_x} \nabla f(x)) - (x - \eta \mathcal{P}_{\mathcal{T}_x} \nabla f(x))\| \\ & + \frac{\beta_L}{2} \|\mathcal{P}_{\mathcal{M}}(x - \eta \mathcal{P}_{\mathcal{T}_x} \nabla f(x)) - (x - \eta \mathcal{P}_{\mathcal{T}_x} \nabla f(x))\|^2 \\ \le\ & \frac{4\eta^2 \|\mathcal{P}_{\mathcal{T}_x} \nabla f(x)\|^2 l}{R} + \frac{16\eta^4 \beta_L \|\mathcal{P}_{\mathcal{T}_x} \nabla f(x)\|^4}{R^2}. \end{aligned}$$

The last line uses that $\|\mathcal{P}_{\mathcal{M}}(x - \eta \mathcal{P}_{\mathcal{T}_x} \nabla f(x)) - (x - \eta \mathcal{P}_{\mathcal{T}_x} \nabla f(x))\| \le 4\|\mathcal{P}_{\mathcal{M}}(x - \eta \mathcal{P}_{\mathcal{T}_x} \nabla f(x)) - (x -$

$\eta\mathcal{P}_{\mathcal{T}_x}\nabla f(x))\|^2/R$ from Lemma 1. So

$$\begin{aligned}
&f(\mathcal{P}_{\mathcal{M}}(x-\eta\mathcal{P}_{\mathcal{T}_x}\nabla f(x)))\\
\le& f(x)-(\eta-\frac{\eta^2}{2\beta_L}-\frac{4\eta^2 l}{R}\\
&-\frac{16\beta_L\eta^4\|\mathcal{P}_{\mathcal{T}_x}\nabla f(x)\|^2}{R^2})\|\mathcal{P}_{\mathcal{T}_x}\nabla f(x)\|^2\\
\le& f(x)-(\eta-\frac{\eta^2}{2\beta_L}-\frac{4\eta^2 l}{R}-\frac{16\eta^4\beta_L l^2}{R^2})\|\mathcal{P}_{\mathcal{T}_x}\nabla f(x)\|^2.
\end{aligned}$$

Let $\eta=\frac{c_{\max}}{\beta_L}\le\frac{1}{80}\min(1,\frac{\beta_L R}{l})/\beta_L$, then when gradient is bigger than ε, with one iteration the function value decreases at least $\frac{1}{4}(c_{\max}\varepsilon^2/\beta_L)$.

3.2. Case 2: Saddle point

Now we observe how the algorithm works at a saddle point on the manifold, where $\|\mathcal{P}_{\mathcal{T}_x}\nabla f(x)\|\le\epsilon$ and $\exists v\in\mathcal{T}_x, v^T\nabla_{xx}L(x,\lambda^*)v\le-\gamma$. For notation simplicity, denote $g_L(x)=\mathcal{P}_{\mathcal{T}_x}\nabla f(x)=\nabla_x L(x,\lambda^*)$, $\nabla_{xx}L(x,\lambda^*)=H_L(x)$. Denote the current iterate by x_{t_0}. Following (Jin et al., 2017a), denote

$$\begin{aligned}
\nu&=\frac{\rho R}{\rho R+8\beta_L},\ \mathscr{H}=\log(\frac{d\beta_L}{\gamma\delta}),\\
\mathscr{F}&=\eta\beta_L\nu\frac{\gamma^3}{\rho^2}\mathscr{H}^{-3},\ \mathscr{G}=\sqrt{\eta\beta_L\nu}\frac{\gamma^2}{\rho}\mathscr{H}^{-2},\\
\mathscr{S}&=\sqrt{\eta\beta_L\nu}\frac{\gamma}{\rho}\mathscr{H}^{-1},\ \mathscr{T}=\frac{\mathscr{H}}{\eta\gamma},\\
\tilde{f}_y(x)&=f(y)+g_L(y)^T(x-y)+\\
&\quad\frac{1}{2}(x-y)^T\mathcal{P}_{\mathcal{T}_{x_{t_0}}}H_L(x_{t_0})\mathcal{P}_{\mathcal{T}_{x_{t_0}}}(x-y).
\end{aligned}$$

We need Lemma 2, which is similar to Lemma 16 and 17 in (Jin et al., 2017a). Here is a sketch of the proof. The first point says that, if the function value is not decreased, then the iterates stay close to the saddle points; the second point says that, if the iterates stay close to the saddle points, then the norm of the movement's projection onto the smallest eigenvector increases, and by contradiction, iterates escape from the saddle point. In the supplement, we prove these two points.

Lemma 2. *Let* $\|g_L(x_{t_0})\|\le\mathscr{G}$ *and* $\lambda_{\min}(\mathcal{P}_{\mathcal{T}_{x_{t_0}}}H_L(x_{t_0})\mathcal{P}_{\mathcal{T}_{x_{t_0}}})\le-\gamma$. *There exists a constant* $c_{\max}$ *such that* $\forall\hat{c}>3,\delta\in(0,\frac{d\kappa}{e}]$, *for any* u_{t_0} *with* $\|u_{t_0}-x_{t_0}\|\le 2\mathscr{S}/(\kappa\log(\frac{d\kappa}{\delta}))$, $\kappa=\beta_L/\gamma$ *the following holds.*

- *Define*

  $$T=\min\left\{\inf_t\left\{t|\tilde{f}_{u_{t_0}}(u_t)-f(u_{t_0})\le-3\mathscr{F}\right\},\hat{c}\mathscr{T}\right\},$$

  then $\forall\eta\le c_{\max}/l$, *we have* $\forall t_0<t<t_0+T$, $\|u_t-x\|\le 200(\hat{c}\mathscr{S})$.

- *Take two points* u_{t_0} *and* w_{t_0} *which are perturbed from the saddle point (by adding noise to each point), where* $\|u_{t_0}-x_{t_0}\|\le r$, $w_{t_0}=u_{t_0}+\mu_0 re_1$, e_1 *is the smallest eigenvector of* $H_L(x_{t_0})$, $r=\frac{\mathscr{S}}{\kappa\log(\frac{d\kappa}{\delta})}$, $\mu_0\in[\delta/(2\sqrt{d}),1]$, *and the algorithm runs two sequences* $\{u_t\}$ *and* $\{w_t\}$ *starting from* u_{t_0} *and* w_{t_0}. *Denote*

  $$T=\min\left\{\inf_t\left\{t|\tilde{f}_{w_{t_0}}(w_t)-f(w_{t_0})\le-3\mathscr{F}\right\},\hat{c}\mathscr{T}\right\},$$

  then $\forall\eta\le c_{\max}/l$, *if* $\forall t_0<t<t_0+T$, $\|u_t-x\|\le 200(\hat{c}\mathscr{S})$, *we have* $T<\hat{c}\mathscr{T}$.

4. Conclusions and Discussion

We have shown that for the constrained optimization problem of minimizing $f(x)$ subject to a manifold constraint (or a set of equality constraints $c_i(x)=0$), as long as the function and the manifold are appropriately smooth, a perturbed projected gradient descent algorithm will escape saddle points with a rate that has an $1/\epsilon^2$ dependence on the accuracy, a polynomial dependence on the curvature and smoothness parameters, and more importantly, a *polylog dependence* on the problem dimension (hence it the number of iterations is almost dimension-free). This improves the rate of the best known result for constrained optimization, which was polynomial in dimension (Ge et al., 2015).

Future work with examine modifying the algorithm in (Tripuraneni et al., 2018) to get an algorithm using the Riemannian gradient. This is not immediate, because the retraction map, as an approximated tangent space and manifold mapping, does not appear to yield a sufficiently tight error bound to prove Lemma 2.

Things become more difficult when the constraints consist of inequalities. The RLICQ assumption used here is no longer applicable. As an example, consider $c_i(x)=x_i$, $i=1,...,d-1$ where $x\in\mathbb{R}^d$. C is identity without the last column, whose smallest singular value is 1. If the constraint is $c_i(x)=0$, then we have a subspace x_d axis, which is easy to optimize over. If the constraint is $c_i(x)\ge 0$, the problem is $\min f(x)$, s.t., $x_i\ge 0$, $i=1,...,d-1$. Finding its local minimum falls into an NP-complete problem (Murty & Kabadi, 1987).

A natural extension of our result is to consider other variants of gradient descent, such as the heavy ball method, Nesterov's acceleration, and the stochastic setting. The question is whether these algorithms with appropriate modification (with manifold constraints), would perform not too differently from the unconstrained case, and whether it is possible show the relationship between convergence rate and smoothness of manifold.

References

Allen-Zhu, Zeyuan. Natasha 2: Faster Non-Convex Optimization Than SGD. *arXiv preprint arXiv:1708.08694*, 2017.

Boumal, Nicolas, Voroninski, Vlad, and Bandeira, Afonso. The non-convex burer-monteiro approach works on smooth semidefinite programs. In *Advances in Neural Information Processing Systems*, pp. 2757–2765, 2016.

Boumal, Nicolas, Absil, P-A, and Cartis, Coralia. Global rates of convergence for nonconvex optimization on manifolds. *IMA Journal of Numerical Analysis*, pp. drx080, 2018. doi: 10.1093/imanum/drx080. URL `http://dx.doi.org/10.1093/imanum/drx080`.

Carmon, Yair and Duchi, John C. Gradient descent efficiently finds the cubic-regularized non-convex newton step. *arXiv preprint arXiv:1612.00547*, 2017.

Du, Simon S, Jin, Chi, Lee, Jason D, Jordan, Michael I, Singh, Aarti, and Poczos, Barnabas. Gradient descent can take exponential time to escape saddle points. In *Advances in Neural Information Processing Systems*, pp. 1067–1077, 2017.

Ge, Rong, Huang, Furong, Jin, Chi, and Yuan, Yang. Escaping from saddle points – online stochastic gradient for tensor decomposition. In *Conference on Learning Theory*, pp. 797–842, 2015.

Jin, Chi, Ge, Rong, Netrapalli, Praneeth, Kakade, Sham M, and Jordan, Michael I. How to escape saddle points efficiently. In *International Conference on Machine Learning*, pp. 1724–1732, 2017a.

Jin, Chi, Netrapalli, Praneeth, and Jordan, Michael I. Accelerated gradient descent escapes saddle points faster than gradient descent. *arXiv preprint arXiv:1711.10456*, 2017b.

Lee, Jason D., Simchowitz, Max, Jordan, Michael I., and Recht, Benjamin. Gradient descent only converges to minimizers. *Conference on Learning Theory*, pp. 1246–1257, 2016.

Murty, Katta G and Kabadi, Santosh N. Some np-complete problems in quadratic and nonlinear programming. *Mathematical programming*, 39(2):117–129, 1987.

Nesterov, Yu and Polyak, Boris T. Cubic regularization of newton method and its global performance. *Mathematical Programming*, 108(1), 2016.

Pemantle, Robin. Nonconvergence to unstable points in urn models and stochastic approximations. *The Annals of Probability*, pp. 698–712, 1990.

Simchowitz, Max, Alaoui, Ahmed El, and Recht, Benjamin. On the gap between strict-saddles and true convexity: An $\Omega(\log \mathrm{d})$ lower bound for eigenvector approximation. *arXiv preprint arXiv:1704.04548*, 2017.

Sun, Ju, Qu, Qing, and Wright, John. Complete dictionary recovery over the sphere. In *Sampling Theory and Applications (SampTA), 2015 International Conference on*, pp. 407–410. IEEE, 2015.

Sun, Ju, Qu, Qing, and Wright, John. A geometric analysis of phase retrieval. In *Information Theory (ISIT), 2016 IEEE International Symposium on*, pp. 2379–2383. IEEE, 2016.

Tripuraneni, Nilesh, Flammarion, Nicolas, Bach, Francis, and Jordan, Michael I. Averaging Stochastic Gradient Descent on Riemannian Manifolds. *arXiv preprint arXiv:1802.09128*, 2018.

Wright, Stephen and Nocedal, Jorge. Numerical optimization. *Springer Science*, 35(67-68):7, 1999.

Zhang, Hongyi, Reddi, Sashank J, and Sra, Suvrit. Riemannian svrg: fast stochastic optimization on riemannian manifolds. In *Advances in Neural Information Processing Systems*, pp. 4592–4600, 2016.