Run-and-Inspect Method for Nonconvex Optimization and Global Optimality Bounds for R-Local Minimizers

Yifan Chen · Yuejiao Sun · Wotao Yin

Abstract Many optimization algorithms converge to stationary points. When the underlying problem is nonconvex, they may get trapped at local minimizers and occasionally stagnate near saddle points. We propose the Run-and-Inspect Method, which adds an "inspect" phase to existing algorithms that helps escape from non-global stationary points. The inspection samples a set of points in a radius R around the current point. When a sample point yields a sufficient decrease in the objective, we move there and resume an existing algorithm. If no sufficient decrease is found, the current point is called an approximate R-local minimizer. We show that an R-local minimizer is globally optimal, up to a specific error depending on R, if the objective function can be implicitly decomposed into a smooth convex function plus a restricted function that is possibly nonconvex, nonsmooth. For high-dimensional problems, we introduce blockwise inspections to overcome the curse of dimensionality while still maintaining optimality bounds up to a factor equal to the number of blocks. Our method performs well on a set of artificial and realistic nonconvex problems by coupling with gradient descent, coordinate descent, EM, and prox-linear algorithms.

Keywords R-local minimizer, Run-and-Inspect Method, nonconvex optimization, global minimum, global optimality

Mathematics Subject Classification (2000) 90C26 · 90C30 · 49M30 · 65K05

1 Introduction

This paper introduces and analyzes *R-local minimizers* in a class of nonconvex optimization and develops a Run-and-Inspect Method to find them.

Consider a possibly nonconvex minimization problem:

$$\min F(\mathbf{x}) = F(x_1, ..., x_s), \tag{1}$$

where the variable $\mathbf{x} \in \mathbb{R}^n$ can be decomposed into s blocks $x_1, ..., x_s$, $s \geq 1$. We assume $x_i \in \mathbb{R}^{n_i}$.

We call a point $\bar{\mathbf{x}}$ an R-local minimizer for some $R > 0$ if it attains the minimum of F within the ball with center $\bar{\mathbf{x}}$ and radius R.

This work of Y. Chen is supported in part by Tsinghua Xuetang Mathematics Program and Top Open Program for his short-term visit to UCLA. The work of Y. Sun and W. Yin is supported in part by NSF grant DMS-1720237 and ONR grant N000141712162.

Yifan Chen
Department of Mathematical Sciences, Tsinghua University, Beijing, China.
E-mail: chenyifan14@mails.tsinghua.edu.cn

Yuejiao Sun · Wotao Yin
Department of Mathematics, University of California, Los Angeles, CA 90095.
E-mail: sunyj / wotaoyin@math.ucla.edu

In nonconvex minimization, it is relatively cheap to find a local minimizer but difficult to obtain a global minimizer. For a given $R > 0$, the difficulty of finding an R-local minimizer lies between those two. Informally, they have the following relationships: for any $R > 0$,

$$\begin{aligned}
&F \text{ is convex } \Rightarrow \\
&\{\text{local minimizers}\} = \{R\text{-local minimizers}\} = \{\text{global minimizers}\}; \\
&F \text{ is nonconvex } \Rightarrow \\
&\{\text{local minimizers}\} \supseteq \{R\text{-local minimizers}\} \supseteq \{\text{global minimizers}\}.
\end{aligned}$$

We are interested in nonconvex problems for which the last "$\supseteq$" holds with "$=$," indicating that any R-local minimizer is global. This is possible, for example, if F is the sum of a quadratic function and a sinusoidal oscillation:

$$F(x) = \frac{x^2}{2} + a \sin\left(b\pi(x - \frac{1}{2b})\right) + a, \tag{2}$$

where $x \in \mathbb{R}$ and $a, b \in \mathbb{R}$. The range of oscillation is specified by amplitude a and frequency $\frac{b}{2}$. We use $-\frac{1}{2b}$ to shift its phase so that the minimizer of F is $x^* = 0$. We also add a to level the minimal objective at $F(x^*) = 0$.

An example of (2) with $a = 0.3$ and $b = 3$ is depicted in Figure 1.

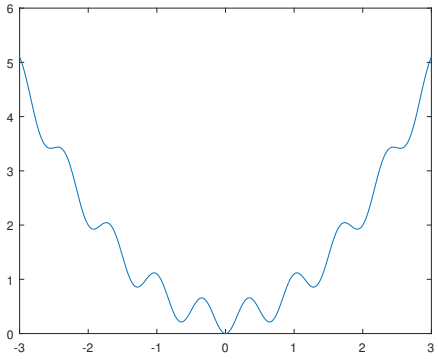

Fig. 1 $F(x)$ in (2) with $a = 0.3,\ b = 3$.

Observe that F has many local minimizers, and its only global minimizer is $x^* = 0$. Near each local minimizer $\bar{x}$, we look for $x \in [\bar{x} - R, \bar{x} + R]$ such that $f(x) < f(\bar{x})$. We claim that by taking $R \geq \min\{2\sqrt{a}, \frac{2}{b}\}$, such x exists for every local minimizer $\bar{x}$ except $\bar{x} = x^*$.

Proposition 1 *Consider minimizing F in (2). If $R \geq \min\{2\sqrt{a}, \frac{2}{b}\}$, then the only point $\bar{x}$ that satisfies the condition*

$$F(\bar{x}) = \min\left\{F(x) : x \in [\bar{x} - R, \bar{x} + R]\right\} \tag{3}$$

is the global minimizer $x^* = 0$.

Proof Suppose $\bar{x} \neq 0$. Without loss of generality we can further assume $\bar{x} > 0$. Recall the global minimizer is $x^* = 0$.
i) If $\bar{x} \leq 2\sqrt{a}$, then $x^* \in [\bar{x} - R, \bar{x} + R]$ gives $F(\bar{x}) = 0$, so $\bar{x}$ is the global minimizer. Otherwise, we have $F(\bar{x} - 2\sqrt{a}) < F(\bar{x})$. Indeed,

$$F(\bar{x} - 2\sqrt{a}) - F(\bar{x}) \leq \frac{(\bar{x} - 2\sqrt{a})^2}{2} - \frac{\bar{x}^2}{2} + 2a = 2\sqrt{a}(2\sqrt{a} - \bar{x}) < 0.$$

However, since $\bar{x} - 2\sqrt{a} \in [\bar{x} - R, \bar{x} + R]$, (3) fails to hold; contradiction.
ii) Similar to part i) above, if $\bar{x} \leq \frac{2}{b}$, then $\bar{x}$ is the global minimizer. Otherwise, we have

$$F(\bar{x} - \tfrac{2}{b}) - F(\bar{x}) = \frac{(\bar{x} - \frac{2}{b})^2}{2} - \frac{\bar{x}^2}{2} < 0.$$

This leads to the contradiction similar to part i). □

Proposition 1 indicates that we can find x^* of this problem by locating an approximate local minimizer $\bar{x}^k$ (using any algorithm) and then inspecting a small region near $\bar{x}^k$ (e.g., by sampling a set of points). Once the inspection finds a point x such that $f(x) < f(\bar{x}^k)$, run the algorithm from x and let it find the next approximate local minimizer $\bar{x}^{k+1}$ such that $f(\bar{x}^{k+1}) \leq f(x)$. Alternate such running and inspection steps until, at a local minimizer $\bar{x}^K$, the inspection fails to find a better point nearby. Then, $\bar{x}^K$ must be an approximate global solution. We call this procedure the *Run-and-Inspect Method*.

The coupling of "run" and "inspect" is simple and flexible because, no matter which point the "run" phase generates, being it a saddle point, local minimizer, or global minimizer, the "inspect" phase will either improve upon it or verify its optimality. Because saddle points are easier to escape from than a non-global local minimizer, hereafter, we ignore saddle points in our discussion. Related saddle-point avoiding algorithms are reviewed below along with other literature.

Inspection by sampling points works in low dimensions. However, it suffers from the curse of dimensionality, as the number of points will increase exponentially in the dimension. For high-dimensional problems, the cost will be prohibitive. To address this issue, we define the blockwise **R**-local minimizer and break the inspection into s blocks of low dimensions: $\mathbf{x} = [x_1^T \; x_2^T \cdots \; x_s^T]^T$ where $x_i \in \mathbb{R}^{n_i}$. We call a point $\bar{\mathbf{x}}$ a *blockwise* $\mathbf{R}$*-local minimizer*, where $\mathbf{R} = [R_1 \; R_2 \; \cdots \; R_s]^T > 0$, if it satisfies

$$F(\bar{x}_1, ..., \bar{x}_i, ..., \bar{x}_s) \leq \min_{x_i \in B(\bar{x}_i, R_i)} F(\bar{x}_1, ..., x_i, ..., \bar{x}_s), \quad \forall 1 \leq i \leq s, \tag{4}$$

where $B(x, R)$ is a closed ball with center x and radius R. To locate a blockwise **R**-local minimizer, the inspection is applied to a block while fixing the others. Its cost grows linearly in the number of blocks when the size of every block is fixed.

This paper studies R-local and blockwise **R**-local minimizers and develop their global optimality bounds for a class of function F that is the sum of a smooth, strongly convex function and a restricted nonconvex function. (Our analysis assumes a property weaker than strong convexity.) Roughly speaking, the landscape of F is convex at a coarse level, but it can have many local minima. (Arguably, if the landscape of F is overall nonconvex, minimizing F is fundamentally difficult.)

This decomposition is implicit and only used to prove bounds. Our Run-and-Inspect Method, which does *not* use the decomposition, can still provably find a solution that has a bounded distance to a global minimizer and an objective value that is bounded by the global minimum. Both bounds can be zero with a finite R.

The radius R affects theoretical bounds, solution quality, and inspection cost. If R is very small, the inspections will be cheap, but the solution returned by our method will be less likely to be global. On the other hand, an excessive large R leads to expensive inspection and is unnecessary since the goal of inspection is to escape local minima rather than decrease the objective. Theoretically, Theorem 3 indicates a proper choice $R = 2\sqrt{\beta/L}$, where β, L are parameters of the functions in the implicit decomposition. Furthermore, if R is larger than a certain threshold given in Theorem 4, then $\bar{\mathbf{x}}$ returned by our method must be a global minimizer. However, as these value and threshold are associated with the implicit decomposition, they are typically unavailable to the user.

One can imagine that a good practical choice of R would be the radius of the global-minimum valley, assuming this valley is larger than all other local-minimum valleys. This choice is hard to guess, too. Another choice of R is roughly inversely proportional to $\|\nabla f\|$, where f is the smooth convex component in the implicit decomposition of F. It is possible to estimate $\|\nabla f\|$ using an area maximum of $\|\nabla F\|$, which itself requires a radius of sampling, unfortunately. ($\|\nabla F\|$ is zero at any local minimizer, so its local value is useless.) However, this result indicates that local minimizers that are far from the global minimizer are easier to escape from.

We empirically observe that it is both fast and reliable to use a large R and sample the ball $B(\bar{\mathbf{x}}, R)$ outside-in, for example, to sample on a set of rings of radius $R, R-\Delta R, R-2\Delta R, \ldots > 0$. In most cases, a point on the first couple of rings is quickly found, and we escape to that point. The smallest ring is almost never sampled except when $\bar{\mathbf{x}}$ is already an (approximate) global minimizer. Although the final inspection around a global minimizer is generally unavoidable, global minimizers in problems such as compressed sensing and matrix decomposition can be identified without inspection because they have the desired structure or attained a lower bound to the objective value. Anyway, it appears that choosing R is ad hoc but not difficult. Throughout our numerical experiments, we use $R = O(1)$ and obtain excellent results consistently.

The exposition of this paper is limited to deterministic methods though it is possible to apply stochastic techniques. We can undoubtedly adopt stochastic approximation in the "run" phase when, for example, the objective function has a large-sum structure. Also, if the problem has a coordinate-friendly structure [16], we can randomly choose a coordinate, or a block of coordinates, to update each time. Another direction worth pursuing is stochastic sampling during the "inspect" phase. These stochastic techniques are attractive in specific settings, but we focus on non-stochastic techniques and global guarantees in this paper.

1.1 Related work

1.1.1 No spurious local minimum

For certain nonconvex problems, a local minimum is always global or good enough. Examples include tensor decomposition [6], matrix completion [7], phase retrieval [22], and dictionary learning [21] under proper assumptions. When those assumptions are violated to a moderate amount, spurious local minima may appear and be possibly easy to escape. We will inspect them in our future work.

1.1.2 First-order methods, derivative-free method, and trust-region method

For nonconvex optimization, there has been recent work on first-order methods that can guarantee convergence to a stationary point. Examples include the block coordinate update method [25], ADMM for nonconvex optimization [23], the accelerated gradient algorithm [8], the stochastic variance reduction method [18], and so on.

Because the "inspect" phase of our method uses a radius, it is seemingly related to the trust-region method [3,12] and derivative-free method [4], both of which also use a radius at each step. However, the latter methods are not specifically designed to escape from a non-global local minimizer.

1.1.3 Avoiding saddle points

A recent line of work aims to avoid saddle points and converge to an ϵ-second-order stationary point $\bar{\mathbf{x}}$ that satisfies

$$\|\nabla F(\bar{\mathbf{x}})\| \le \epsilon \quad \text{and} \quad \lambda_{\min}(\nabla^2 F(\bar{\mathbf{x}})) \ge -\sqrt{\rho\epsilon}, \tag{5}$$

where ρ is the Lipschitz constant of $\nabla^2 F(\mathbf{x})$. Their assumption is the *strict saddle* property, that is, a point satisfying (5) for some $\rho > 0$ and $\epsilon > 0$ must be an approximate local minimizer. On the algorithmic side, there are second-order algorithms [13,15] and first-order stochastic methods [6,9,14] that can escape saddle points. The second-order algorithms use Hessian information and thus are more expensive at each iteration in high dimensions. Our method can also avoid saddle points.

1.1.4 Simulated annealing

Simulated annealing (SA) [11] is a classical method in global optimization, and thermodynamic principles can interpret it. SA uses a Markov chain with a stationary distribution $\sim e^{-\frac{F(\mathbf{x})}{T}}$, where T is the temperature parameter. By decreasing T, the distribution tends to concentrate on the global minimizer of $F(\mathbf{x})$. However, it is difficult to know exactly when it converges, and the convergence rate can be extremely slow.

SA can be also viewed as a method that samples the Gibbs distribution using Markov-Chain Monte Carlo (MCMC). Hence, we can apply SA in the "inspection" of our method. SA will generate more samples in a preferred area that are more likely to contain a better point, which once found will stop the inspection. Apparently, because of the hit-and-run nature of our inspection, we do not need to wait for the SA dynamic to converge.

1.1.5 Flat minima in the training of neural networks

Training a (deep) neural network involves nonconvex optimization. We do not necessarily need to find a global minimizer. A local minimizer will suffice if it generalizes well to data not used in training. There are many recent attempts [1,2,19] that investigate the optimization landscapes and propose methods to find local minima sitting in "rather flat valleys."

Paper [1] uses entropy-SGD iteration to favor flatter minima. It can be seen as a PDE-based smoothing technique [2], which shows that the optimization landscape becomes flatter after smoothing. It makes the theoretical analysis easier and provides explanations for many interesting phenomena in deep neural networks. But, as [24] has suggested, a better non-local quantity is required to go further.

1.2 Notation

Throughout the paper, $\|\cdot\|$ denotes the Euclidean norm. Boldface lower-case letters (e.g., $\mathbf{x}$) denote vectors. However, when a vector is a block in a larger vector, it is represented with a lower-case letter with a subscript, e.g., x_i.

1.3 Organization

The rest of this paper is organized as follows. Section 2 presents the main analysis of R-local and blockwise $\mathbf{R}$-local minimizers, and then introduces the Run-and-Inspect Method. Section 3 presents numerical results of our Run-and-Inspect method. Finally, Section 4 concludes this paper.

2 Main Results

In sections 2.1–2.3, we develop theoretical guarantees for our R-local and $\mathbf{R}$-local minimizers for a class of nonconvex problems. Then, in section 2.4, we design algorithms to find those minimizers.

2.1 Global optimality bounds

In this section, we investigate an approach toward deriving error bounds for a point with certain properties.

Consider problem (1), and let $\mathbf{x}^*$ denote one of its global minimizers. A global minimizer owns many nice properties. Finding a global minimizer is equivalent to finding a point satisfying all these properties. Clearly, it is easier to develop algorithms that aim at finding a point $\bar{\mathbf{x}}$ satisfying only some of those properties. An example is that when F is everywhere differentiable, $\nabla F(\mathbf{x}^*) = 0$ is a necessary optimality condition. So, many first-order algorithms that produce a sequence $\mathbf{x}^k$ such that $\|\nabla F(\mathbf{x}^k)\| \to 0$ *may*

converge to a global minimizer. Below, we focus on choosing the properties of $\mathbf{x}^*$ so that a point $\bar{\mathbf{x}}$ satisfying the same properties will enjoy bounds on $F(\bar{\mathbf{x}}) - F(\mathbf{x}^*)$ and $\|\bar{\mathbf{x}} - \mathbf{x}^*\|$. Of course, proper assumptions on F are needed, which we will make as we proceed.

Let us use $\mathbf{Q}$ to represent a certain set of properties of $\mathbf{x}^*$, and define

$$S_{\mathbf{Q}} = \{\mathbf{x} : \mathbf{x} \text{ satisfies property } \mathbf{Q}\}, \tag{6}$$

which includes $\mathbf{x}^*$. For any point $\bar{\mathbf{x}}$ that also belongs to the set, we have

$$F(\bar{\mathbf{x}}) - F(\mathbf{x}^*) \leq \max_{\mathbf{x},\mathbf{y} \in S_{\mathbf{Q}}} F(\mathbf{x}) - F(\mathbf{y})$$

and

$$\|\bar{\mathbf{x}} - \mathbf{x}^*\| \leq \operatorname{diam}(S_{\mathbf{Q}}),$$

where $\operatorname{diam}(S_{\mathbf{Q}})$ stands for the diameter of the set $S_{\mathbf{Q}}$. Hence, the problem of constructing an error bound reduces to analyzing the set $S_{\mathbf{Q}}$ under certain assumptions on F.

As an example, consider a μ-strongly convex and differentiable F and a simple choice of $\mathbf{Q}$ as $\|\nabla F(\mathbf{x})\| \leq \delta$ with $\delta > 0$. This choice is admissible since $\|\nabla F(\mathbf{x}^*)\| = 0 \leq \delta$. For this choice, we have

$$F(\bar{\mathbf{x}}) - F(\mathbf{x}^*) \leq \frac{\|\nabla F(\bar{\mathbf{x}})\|^2}{2\mu} \leq \frac{\delta^2}{2\mu},$$

and

$$\|\bar{\mathbf{x}} - \mathbf{x}^*\| \leq \frac{\|\nabla F(\bar{\mathbf{x}})\|}{\mu} \leq \frac{\delta}{\mu},$$

where the first "$\leq$" in both bounds follows from the strong convexity of F.

We now restrict F to the implicit decomposition

$$\boxed{F(\mathbf{x}) = f(\mathbf{x}) + r(\mathbf{x}).} \tag{7}$$

We use the term "implicit" because this decomposition is only used for analysis, not required by our Run-and-Inspect Method. Define the sets of the global minimizers of F and f as, respectively,

$$\chi^* := \{\mathbf{x} : F(\mathbf{x}) = \min_{\mathbf{y}} F(\mathbf{y})\},$$
$$\chi_f^* := \{\mathbf{x} : f(\mathbf{x}) = \min_{\mathbf{y}} f(\mathbf{y})\}.$$

Below we make three assumptions on (7). The first and third assumptions are used throughout this section. Only some of our results require the second assumption.

Assumption 1 *$f(\mathbf{x})$ is differentiable, and $\nabla f(\mathbf{x})$ is L-Lipschitz continuous.*

Assumption 2 *$f(\mathbf{x})$ satisfies the Polyak-Łojasiewicz (PL) inequality [17] with $\mu > 0$:*

$$\frac{1}{2}\|\nabla f(\mathbf{x})\|^2 \geq \mu(f(\mathbf{x}) - f(\mathbf{x}^*)), \quad \forall \mathbf{x} \in \mathbb{R}^n,\ \mathbf{x}^* \in \chi_f^*. \tag{8}$$

Given a point $\mathbf{x}$, we define its projection

$$\mathbf{x}_{\mathrm{P}} := \operatorname*{argmin}_{\mathbf{x}^* \in \chi_f^*} \{\|\mathbf{x}^* - \mathbf{x}\|\}.$$

Then, the PL inequality (8) yields the quadratic growth (QG) condition [10]:

$$f(\mathbf{x}) - f(\mathbf{x}^*) = f(\mathbf{x}) - f(\mathbf{x}_{\mathrm{P}}) \geq \frac{\mu}{2}\|\mathbf{x} - \mathbf{x}_{\mathrm{P}}\|^2, \quad \forall \mathbf{x} \in \mathbb{R}^n. \tag{9}$$

Clearly, (8) and (9) together imply

$$\|\nabla f(\mathbf{x})\| \geq \mu\|\mathbf{x} - \mathbf{x}_{\mathrm{P}}\|. \tag{10}$$

Assumption 2 ensures that the gradient of f bounds its objective error.

Assumption 3 *$r(\mathbf{x})$ satisfies $|r(\mathbf{x}) - r(\mathbf{y})| \le \alpha\|\mathbf{x}-\mathbf{y}\| + 2\beta$ in which $\alpha, \beta \ge 0$ are constants.*

Assumption 3 implies that r is overall α-Lipschitz continuous with additional oscillations up to 2β. In the implicit decomposition (7), though r can cause F to have non-global local minimizers, its impact on the overall landscape of f is limited. For example, the ℓ_p^p $(0 < p < 1)$ penalty in compressed sensing is used to induce sparsity of solutions. It is nonconvex and satisfies our assumption

$$|x|^p - |y|^p \le \big||x| - |y|\big|^p \le p|x-y| + 1 - p, \quad x, y \in \mathbb{R}.$$

In fact, many sparsity-induced penalties satisfy Assumption 3. Many of them are sharp near 0 and thus not Lipschitz there. In Assumption 3, β models their variation near 0 and α controls their increase elsewhere.

In section 2.2, we will show that every $\mathbf{x}^* \in \chi^*$ satisfies $\|\nabla f(\mathbf{x}^*)\| \le \delta$ for a universal δ that depends on α, β, L. So, we choose the condition

$$\boxed{\mathbf{Q}: \ \|\nabla f(\mathbf{x})\| \le \delta.} \tag{11}$$

To derive the error bound, we introduce yet another assumption:

Assumption 4 *The set χ_f^* is bounded. That is, there exists $M \ge 0$ such that, for any $\mathbf{x}, \mathbf{y} \in \chi_f^*$, we have $\|\mathbf{x}-\mathbf{y}\| \le M$.*

When f has a unique global minimizer, we have $M = 0$ in Assumption 4.

Theorem 2 *Take Assumptions 1, 2 and 3, and assume that all points in χ^* have property $\mathbf{Q}$ in (11). Then, the following properties hold for every $\bar{\mathbf{x}} \in S_{\mathbf{Q}}$:*

1. *$F(\bar{\mathbf{x}}) - F^* \le \frac{\delta^2}{2\mu} + 2\beta$, if $\alpha = 0$ in Assumption 3;*
2. *$d(\bar{\mathbf{x}}, \chi^*) \le \frac{2\delta}{\mu} + M$ and $F(\bar{\mathbf{x}}) - F^* \le \frac{\delta^2 + 2\alpha\delta}{\mu} + \alpha M + 2\beta$, if $\alpha \ge 0$ and Assumption 4 holds.*

Proof To show part 1, we have

$$\begin{aligned} F(\bar{\mathbf{x}}) - F^* &= (f(\bar{\mathbf{x}}) - f(\mathbf{x}^*)) + (r(\bar{\mathbf{x}}) - r(\mathbf{x}^*)) \le \max_{\mathbf{x}\in\mathbb{R}^n} (f(\bar{\mathbf{x}}) - f(\mathbf{x})) + 2\beta \\ &\overset{(8)}{\le} \frac{\|\nabla f(\bar{\mathbf{x}})\|^2}{2\mu} + 2\beta \le \frac{\delta^2}{2\mu} + 2\beta. \end{aligned}$$

Part 2: Since f satisfies the PL inequality (8) and $\bar{\mathbf{x}} \in S_{\mathbf{Q}}$, we have

$$d(\bar{\mathbf{x}}, \chi_f^*) \overset{(10)}{\le} \frac{\|\nabla f(\bar{\mathbf{x}})\|}{\mu} \overset{(11)}{\le} \frac{\delta}{\mu}.$$

By choosing an $\mathbf{x}^* \in \chi^*$ and noticing $\mathbf{x}^* \in S_{\mathbf{Q}}$, we also have $d(\mathbf{x}^*, \chi_f^*) \le \frac{\delta}{\mu}$ and thus

$$d(\bar{\mathbf{x}}, \chi^*) \le d(\bar{\mathbf{x}}, \chi_f^*) + M + d(x^*, \chi_f^*) \le \frac{2\delta}{\mu} + M.$$

Below we let $\bar{\mathbf{x}}_{\mathrm{P}}$ and $\mathbf{x}_{\mathrm{P}}^*$, respectively, denote the projections of $\bar{\mathbf{x}}$ and $\mathbf{x}^*$ onto the set χ_f^*. Since $f(\bar{\mathbf{x}}_{\mathrm{P}}) = f(\mathbf{x}_{\mathrm{P}}^*)$, we obtain

$$\begin{aligned} F(\bar{\mathbf{x}}) - F^* &= (f(\bar{\mathbf{x}}) - f(\bar{\mathbf{x}}_{\mathrm{P}})) + \big(f(\mathbf{x}_{\mathrm{P}}^*) - f(\mathbf{x}^*)\big) + \big(r(\bar{\mathbf{x}}) - r(\mathbf{x}^*)\big) \\ &\le \frac{\|\nabla f(\bar{\mathbf{x}})\|^2}{2\mu} + \frac{\|\nabla f(\mathbf{x}^*)\|^2}{2\mu} + \alpha\|\bar{\mathbf{x}} - \mathbf{x}^*\| + 2\beta \\ &\le \frac{\delta^2 + 2\alpha\delta}{\mu} + \alpha M + 2\beta. \end{aligned}$$

□

In the theorem above, we have constructed global optimality bounds for $\bar{\mathbf{x}}$ obeying $\mathbf{Q}$. In the next two subsections, we show that R-local minimizers, which include global minimizers, do obey $\mathbf{Q}$ under mild conditions. Hence, the bounds apply to any R-local minimizer.

2.2 R-local minimizers

In this section, we define and analyze R-local minimizers. We discuss its blockwise version in section 2.3. Throughout this subsection, we assume that $R \in (0, \infty]$, and $B(\mathbf{x}, R)$ is a *closed* ball centered at $\mathbf{x}$ with radius R.

Definition 1 *The point $\bar{\mathbf{x}}$ is called a standard R-local minimizer of F if it satisfies*

$$F(\bar{\mathbf{x}}) = \min_{\mathbf{x} \in B(\bar{\mathbf{x}}, R)} F(\mathbf{x}). \tag{12}$$

Obviously an R-local minimizer is a local minimizer, and when $R = \infty$, it is a global minimizer. Conversely, a global minimizer is always an R-local minimizer.

We first bound the gradient of f at an R-local minimizer so that $\mathbf{Q}$ in (11) is satisfied.

Theorem 3 *Suppose, in (7), f and r satisfy Assumptions 1 and 3. Then, a point $\bar{\mathbf{x}}$ obeys $\mathbf{Q}$ in (11) with δ given in the following two cases:*

1. *$\delta = \alpha$ if r is differentiable with $\alpha \geq 0$ and $\beta = 0$ in (3) and $\bar{\mathbf{x}}$ is a stationary point of F;*
2. *$\delta = \alpha + \max\{\frac{4\beta}{R}, 2\sqrt{\beta L}\}$ if $\bar{\mathbf{x}}$ is a standard R-local minimizer of F.*

Proof Under the conditions in part 1, we have $\beta = 0$ and $\nabla F(\bar{\mathbf{x}}) = 0$, so $\|\nabla f(\bar{\mathbf{x}})\| = \|\nabla r(\bar{\mathbf{x}})\| \leq \alpha = \delta$. Hence, $\mathbf{Q}$ is satisfied.

Under the conditions in part 2, $\bar{\mathbf{x}}$ is an R-local minimizer of F; hence,

$$\begin{aligned}
&\min_{\mathbf{x} \in B(\bar{\mathbf{x}}, R)} \{f(\mathbf{x}) - f(\bar{\mathbf{x}}) + r(\mathbf{x}) - r(\bar{\mathbf{x}})\} \geq 0, \\
\overset{a)}{\Rightarrow} &\min_{\mathbf{x} \in B(\bar{\mathbf{x}}, R)} \{2\beta + \alpha\|\mathbf{x} - \bar{\mathbf{x}}\| + \langle \nabla f(\bar{\mathbf{x}}), \mathbf{x} - \bar{\mathbf{x}} \rangle + \frac{L}{2}\|\mathbf{x} - \bar{\mathbf{x}}\|^2\} \geq 0, \\
\overset{b)}{\Leftrightarrow} &\min_{\mathbf{x} \in B(\bar{\mathbf{x}}, R)} \{2\beta + (\alpha - \|\nabla f(\bar{\mathbf{x}})\|)\|\mathbf{x} - \bar{\mathbf{x}}\| + \frac{L}{2}\|\mathbf{x} - \bar{\mathbf{x}}\|^2\} \geq 0,
\end{aligned} \tag{13}$$

where $a)$ is due to the assumption on r and that $\nabla f(\mathbf{x})$ is L-Lipschitz continuous; $b)$ is because, as $\|\mathbf{x} - \bar{\mathbf{x}}\|$ is fixed, the minimum is attained with $\frac{\mathbf{x} - \bar{\mathbf{x}}}{\|\mathbf{x} - \bar{\mathbf{x}}\|} = -\frac{\nabla f(\mathbf{x})}{\|\nabla f(\mathbf{x})\|}$. If $\|\nabla f(x)\| \leq \alpha$, $\mathbf{Q}$ is immediately satisfied. Now assume $\|\nabla f(x)\| > \alpha$. To simplify (13), we only need to minimize a quadratic function of $\|\mathbf{x} - \bar{\mathbf{x}}\|$ over $[0, R]$. Hence, the objective equals

$$\begin{cases} 2\beta + (\alpha - \|\nabla f(\bar{\mathbf{x}})\|)R + \frac{L}{2}R^2, & \text{if } R \leq \frac{\|\nabla f(\bar{\mathbf{x}})\| - \alpha}{L}, \\ 2\beta - \frac{(\|\nabla f(\bar{\mathbf{x}})\| - \alpha)^2}{2L}, & \text{otherwise.} \end{cases} \tag{14}$$

If $R \leq \frac{\|\nabla f(\bar{\mathbf{x}})\| - \alpha}{L}$, from $2\beta + (\alpha - \|\nabla f(\bar{\mathbf{x}})\|)R + \frac{L}{2}R^2 \geq 0$, we get

$$\begin{aligned}
&\|\nabla f(\bar{\mathbf{x}})\| \leq \alpha + \frac{2\beta}{R} + \frac{LR}{2} \leq \alpha + \frac{2\beta}{R} + \frac{\|\nabla f(\bar{\mathbf{x}})\| - \alpha}{2} \\
\Rightarrow &\|\nabla f(\bar{\mathbf{x}})\| \leq \alpha + \frac{4\beta}{R}.
\end{aligned}$$

Otherwise, from $2\beta - \frac{(\|\nabla f(\bar{\mathbf{x}})\| - \alpha)^2}{2L} \geq 0$, we get

$$\|\nabla f(\bar{\mathbf{x}})\| \leq \alpha + 2\sqrt{\beta L}.$$

Combining both cases, we have $\|\nabla f(\bar{\mathbf{x}})\| \leq \alpha + \max\{\frac{4\beta}{R}, 2\sqrt{\beta L}\} = \delta$ and thus $\mathbf{Q}$. □

The next result is a consequence of part 2 of the theorem above. It presents the values of R that ensure the escape from a non-global local minimizer. In addition, more distant local minimizers $\mathbf{x}$ are easier to escape in the sense that R is roughly inversely proportional to $\|\nabla f(\mathbf{x})\|$.

Corollary 1 *Let $\mathbf{x}$ be a local minimizer of F and $\|\nabla f(\mathbf{x})\| > \alpha + 2\sqrt{\beta L}$. As long as either $R > \frac{4\beta}{\|\nabla f(\mathbf{x})\| - \alpha}$ or $R \geq 2\sqrt{\beta/L}$, there exists $\mathbf{y} \in B(\mathbf{x}, R)$ such that $F(\mathbf{y}) < F(\mathbf{x})$.*

Proof Assume that the result does *not* hold. Then, $\mathbf{x}$ is an R-local minimizer of F. Applying part 2 of Theorem 3, we get $\|\nabla F(\mathbf{x})\| \leq \delta = \alpha + \max\{\frac{4\beta}{R}, 2\sqrt{\beta L}\}$. Combining this with the assumption $\|\nabla f(\mathbf{x})\| > \alpha + 2\sqrt{\beta L}$, we obtain

$$\alpha + 2\sqrt{\beta L} < \|\nabla f(\mathbf{x})\| \leq \alpha + \max\{\frac{4\beta}{R}, 2\sqrt{\beta L}\},$$

from which we conclude $2\sqrt{\beta L} < \frac{4\beta}{R}$ and $\|\nabla f(\mathbf{x})\| \leq \alpha + \frac{4\beta}{R}$; We have reached a contradiction. □

We can further increase R to ensure that any R-local minimizer $\bar{\mathbf{x}}$ is a global minimizer.

Theorem 4 *Under Assumptions 1, 2 and 3 and $R \geq 2\sqrt{\beta/L}$, we have $d(\bar{\mathbf{x}}, \chi^*) \leq 2\frac{\alpha+2\sqrt{\beta L}}{\mu} + M$ for any R-local minimizer $\bar{\mathbf{x}}$. Therefore, if $R \geq 2\frac{\alpha+2\sqrt{\beta L}}{\mu} + M$, any R-local minimizer $\bar{\mathbf{x}}$ is a global minimizer.*

Proof According to Theorem 2, part 2, and Theorem 3, part 2,

$$d(\bar{\mathbf{x}}, \chi^*) \leq 2\frac{\delta}{\mu} + M \leq 2\frac{\alpha + 2\sqrt{\beta L}}{\mu} + M,$$

where, for the latter inequality, we have used $R \geq 2\sqrt{\beta/L}$ and thus $\max\{4\beta/R, 2\sqrt{\beta L}\} = 2\sqrt{\beta L}$. By convex analysis on f, we have $\mu \leq L$. Using it with $\alpha \geq 0$ and $M \geq 0$, we further get $2\frac{\alpha+2\sqrt{\beta L}}{\mu} + M \geq 4\sqrt{\beta L}/\mu \geq 4\sqrt{\beta L}/L \geq 2\sqrt{\beta/L}$. Therefore, if $R \geq 2\frac{\alpha+2\sqrt{\beta L}}{\mu} + M$, then there exists $\mathbf{x}^* \in \chi^*$ such that $\mathbf{x}^* \in B(\bar{\mathbf{x}}, R)$. Being an R-local minimizer means $\bar{\mathbf{x}}$ satisfies $F(\bar{\mathbf{x}}) \leq F(\mathbf{x}^*)$, so $\bar{\mathbf{x}}$ is a global minimizer. □

Remark 1 *Since the decomposition (7) is implicit, the constants in our analysis are difficult to estimate in practice. However, if we have a rough estimate of the distance between the global minimizer and its nearby local minimizers, then this distance appears to be a good empirical choice for R.*

2.3 Blockwise **R**-local minimizers

In this section, we focus on problem (1). This blockwise structure of F motivates us to consider blockwise algorithms. Suppose $\mathbf{R} \in \mathbb{R}^s$ and $\mathbf{R} = (R_1, ..., R_s) \geq 0$. When we fix all blocks but x_i, we write $F(\bar{x}_1, ..., x_i, ..., \bar{x}_s)$ as $F(x_i, \bar{\mathbf{x}}_{-i})$.

Definition 2 *A point $\bar{\mathbf{x}}$ is called a blockwise $\mathbf{R}$-local minimizer of F if it satisfies*

$$F(\bar{x}_i, \bar{\mathbf{x}}_{-i}) = \min_{x_i \in B(\bar{x}_i, R_i)} F(x_i, \bar{\mathbf{x}}_{-i}), \quad 1 \leq i \leq s,$$

where $F(\bar{\mathbf{x}}) = F(\bar{x}_i, \bar{\mathbf{x}}_{-i})$.

When $\mathbf{R} = \infty$, $\bar{\mathbf{x}}$ is known as a Nash equilibrium point of F.

We can obtain similar estimates on the gradient of f for blockwise R-local minimizers. Recall that $S_{\mathbf{Q}} = \{\mathbf{x} : \|\nabla f(\mathbf{x})\| \leq \delta\}$.

Theorem 5 *Suppose f and r satisfy Assumptions 1 and 3. If $\bar{\mathbf{x}}$ is a blockwise $\mathbf{R}$-local minimizer of F, then $\bar{\mathbf{x}} \in S_{\mathbf{Q}}$ (i.e, the property $\mathbf{Q}$ is met) for $\delta = \|\mathbf{v}\| := (\sum |v_i^2|)^{\frac{1}{2}}$ where $v_i := \alpha + \max\{\frac{4\beta}{R_i}, 2\sqrt{\beta L}\}$, $1 \leq i \leq s$.*

Proof $\bar{x}_i$ is an R_i-local minimizer of $F(x_i, \bar{\mathbf{x}}_{-i})$. Since $F(x_i, \bar{\mathbf{x}}_{-i}) = f(x_i, \bar{\mathbf{x}}_{-i}) + r(x_i, \bar{\mathbf{x}}_{-i})$ and $f(x_i, \bar{\mathbf{x}}_{-i})$ and $r(x_i, \bar{\mathbf{x}}_{-i})$ satisfy Assumption 1 and Assumption 3, Theorem 3 shows that $\|\nabla_i f(\bar{x}_i, \bar{\mathbf{x}}_{-i})\| \leq \alpha + \max\{\frac{4\beta}{R_i}, 2\sqrt{\beta L}\} = v_i$. Hence $\|\nabla f(\bar{\mathbf{x}})\| \leq \|\mathbf{v}\|$. □

Remark 2 *We can also obtain a simplified version of Theorem 5, which is*

$$\|\nabla f(\bar{\mathbf{x}})\| \leq \delta := \sqrt{s}\left(\alpha + \max\{\frac{4\beta}{\min_i R_i}, 2\sqrt{\beta L}\}\right).$$

The main difference between the standar and blockwise estimates is the extra factor $\sqrt{s}$ in the latter.

Remark 3 *Since we can set $R = \infty$, our results apply to Nash equilibrium points.*

Generalized from Corollary 1, the following result provides estimates of R_i for escaping from non-global local minimizers. The estimates are smaller when $\nabla_i f$ are larger.

Corollary 2 *Let $\mathbf{x}$ be a local minimizer of F and $\|\nabla_i f(x_i, \mathbf{x}_{-i})\| > \alpha + 2\sqrt{\beta L}$ for some i. As long as $R_i > \frac{4\beta}{\|\nabla_i f(x_i,\mathbf{x}_{-i})\|-\alpha}$, there exists $y \in B(x_i, R_i)$, such that $F(y, \mathbf{x}_{-i}) < F(x_i, \mathbf{x}_{-i})$.*

The theorem below, which follows from Theorems 2 and 5, bounds the distance of an **R**-local minimizer to the set of global minimizers. We do not have a vector **R** to ensure the global optimality of $\bar{\mathbf{x}}$ due to the blockwise limitation. Of course, after reaching $\bar{\mathbf{x}}$, if we switch to standard (non-blockwise) inspection to obtain an standard R-local minimizer, we will be able to apply Theorem 4.

Theorem 6 *Suppose f and r satisfy Assumptions 1–3. If $\bar{\mathbf{x}}$ is a blockwise **R**-local minimizer of F, then*

$$d(\bar{\mathbf{x}}, \chi^*) \le \tfrac{2\sqrt{s}}{\mu}\left(\alpha + \max\{\frac{4\beta}{\min_i R_i}, 2\sqrt{\beta L}\}\right) + M.$$

2.4 Run-and-Inspect Method

In this section, we introduce our Run-and-Inspect Method. The "run" phase can use any algorithm that monotonically converges to an approximate stationary point. When the algorithm stops at either an approximate local minimizer or a saddle point, our method starts its "inspection" phase, which either moves to a strictly better point or verifies that the current point is an approximate (blockwise) R-local minimizer.

2.4.1 Approximate R-local minimizers

We define *approximate* R-local minimizers. Since an R-local minimizer is a special case of a blockwise **R**-local minimizer, we only deal with the latter. Let $\mathbf{x} = [x_1^T \cdots x_s^T]^T$. A point $\bar{\mathbf{x}}$ is called a blockwise **R**-local minimizer of F *up to* $\eta = [\eta_1 \cdots \eta_s]^T \ge 0$ if it satisfies

$$F(\bar{x}_i, \bar{\mathbf{x}}_{-i}) \le \min_{x_i \in B(\bar{x}_i, R_i)} F(x_i, \bar{\mathbf{x}}_{-i}) + \eta_i, \quad 1 \le i \le s;$$

when $s = 1$, we say $\bar{\mathbf{x}}$ is an R-local minimizer of F up to η. It is easy to modify the proof of Theorem 3 to get:

Theorem 7 *Suppose f and r satisfy Assumptions 1 and 3. Then $\bar{\mathbf{x}} \in S_{\mathbf{Q}}$ if $\bar{\mathbf{x}}$ is a blockwise **R**-local minimizer of F up to η and $\delta \ge \|\mathbf{v}\| := (\sum |v_i^2|)^{\frac{1}{2}}$ for $v_i = \alpha + \max\{\frac{4\beta+2\eta_i}{R_i}, \sqrt{(4\beta + 2\eta_i)L}\}$, $1 \le i \le s$.*

Whenever the condition $\bar{\mathbf{x}} \in S_{\mathbf{Q}}$ holds, our previous results for blockwise **R**-local minimizers are applicable.

2.4.2 Algorithms

Now we present two algorithms based on our Run-and-Inspect Method. Suppose that we have implemented an algorithm and it returns a point $\bar{\mathbf{x}}$. For simplicity let **Alg** denote this algorithm. To verify the global optimality of $\bar{\mathbf{x}}$, we seek to inspect F around $\bar{\mathbf{x}}$ by sampling some points. Since a global search is apparently too costly, the inspection is limited in a ball centered at $\bar{\mathbf{x}}$, and for high-dimensional problems, further limited to lower-dimensional balls.

The inspection strategy is to sample some points in the ball around the current point and stop whenever either a better point is found or it finishes the last point. By "better", we mean the objective value decreases by at least a constant amount $\nu > 0$. We call this ν descent threshold. If a better point is found, we resume **Alg** at that point. If no better point is found, the current point is an R-local or **R**-local minimizer of F up to η, where η depends on the density of sample points and the Lipschitz constant of F in the ball.

Algorithm 1 Run-and-Inspect Method

Set $k = 0$ and choose $\mathbf{x}^0 \in \mathbb{R}^n$;
Choose the descent threshold $\nu > 0$;
loop
  $\bar{\mathbf{x}}^k = \mathbf{Alg}(\mathbf{x}^k)$;
  Generate a set $\mathcal{S}$ of sample points in $B(\bar{\mathbf{x}}^k, R)$;
  if there exists $\mathbf{y} \in \mathcal{S}$ such that $F(\mathbf{y}) < F(\bar{\mathbf{x}}^k) - \nu$ **then**
    $\mathbf{x}^{k+1} = \mathbf{y}$;
    $k = k + 1$;
  else
    stop and **return** $\bar{\mathbf{x}}^k$;
  end if
end loop

If **Alg** is a descent method, i.e., $F(\bar{\mathbf{x}}^k) \leq F(\mathbf{x}^k)$, algorithm 1 will stop and output a point $\bar{\mathbf{x}}^{k^*}$ within finitely many iterations: $k^* \leq \frac{F(\mathbf{x}_0)-F^*}{\nu}$, where F^* is the global minimum of F.

The sampling step is a *hit-and-run*, that is, points are only sampled when they are used, and the sampling stops whenever a better point is obtained (or all points have been used). The method of sampling and the number of sample points can vary throughout iterations and depend on the problem structure. In general, sampling points from the outside toward the inside is more efficient.

Here, we analyze a simple approach in which sufficiently many well-distributed points are sampled to ensure that $\bar{\mathbf{x}}^{k^*}$ is an approximate R-local minimizer.

Theorem 8 *Assume that $F(\mathbf{x})$ is $\bar{L}$-Lipschitz continuous in the ball $B(\bar{\mathbf{x}}, R)$, and the set of sample points $\mathcal{S} = \{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_m\}$ has density r, that is,*

$$\max_{\mathbf{x} \in B(\bar{\mathbf{x}}, R)} \min_{0 \leq j \leq m} \|\mathbf{x} - \mathbf{y}_j\| \leq r$$

where $\mathbf{y}_0 = \bar{\mathbf{x}}$. If

$$F(\mathbf{y}_j) \geq F(\bar{\mathbf{x}}) - \nu, \quad j = 1, 2, \ldots, m,$$

then the point $\bar{\mathbf{x}}$ is an R-local minimizer of F up to $\eta = \nu + \bar{L}r$.

Proof We have for

$$\max_{\mathbf{x} \in B(\bar{\mathbf{x}}, R)} \min_{0 \leq j \leq m} \|F(\mathbf{x}) - F(\mathbf{y}_j)\| \leq \bar{L}r.$$

Since $F(\mathbf{y}_j) > F(\bar{\mathbf{x}}) - \nu$ for all j, it follows

$$F(\bar{\mathbf{x}}) \leq \min_{\mathbf{x} \in B(\bar{\mathbf{x}}, R)} F(\mathbf{x}) + \nu + \bar{L}r.$$

□

When the dimension of $\mathbf{x}$ is high , it is impractical to inspect over a high-dimensional ball. This motivates us to extend algorithm 1 to its blockwise version.

Algorithm 2 Run-and-Inspect Method (blockwise version)

Set $k = 0$ and choose $\mathbf{x}^0 \in \mathbb{R}^n$;
Choose the descent threshold $\nu > 0$;
loop
  $\bar{\mathbf{x}}^k = \mathbf{Alg}(\mathbf{x}^k)$;
  Generate sets $\mathcal{S}_i$ of sample points in $B(\bar{x}_i^k, R_i)$ for $i = 1, ..., s$;
  if there exist i and $z \in \mathcal{S}_i$ such that $F(z, \bar{\mathbf{x}}_{-i}^k) < F(\bar{x}^k, \bar{\mathbf{x}}_{-i}^k) - \nu$ **then**
    $x_i^{k+1} = z$;
    $x_j^{k+1} = \bar{x}_j^k$ for all $j \neq i$;
    $k = k + 1$;
  else
    stop and **return** $\bar{\mathbf{x}}^k$;
  end if
end loop

Algorithm 2 samples points in a block while keeping other block variables fixed. This algorithm ends with an approximate blockwise **R**-local minimizer.

Theorem 9 *Assume that $F(x_i, \bar{\mathbf{x}}_{-i})$ is $\bar{L}_i$-Lipschitz continuous in the ball $B(\bar{x}_i, R_i)$ for $1 \le i \le s$, and the set of sample points $\mathcal{S}_i = \{z_{i1}, z_{i2}, \ldots, z_{im_i}\}, 1 \le i \le s$, has density r blockwisely,*

$$\max_{x_i \in B(\bar{x}_i, R_i)} \min_{0 \le j \le m_i} \|x_i - z_{ij}\| \le r, \quad \forall 1 \le i \le s$$

where $z_{i0} = \bar{x}_i$. If

$$F(z_{ij}, \bar{\mathbf{x}}_{-i}) \ge F(\bar{\mathbf{x}}) - \nu, \quad j = 1, 2, \ldots, m_i, i = 1, 2 \ldots, s,$$

*then $\bar{\mathbf{x}}$ is a blockwise **R**-local minimizer of F up to $\eta = \nu + \bar{L}r$.*

The proof is similar to that of Theorem 8.

3 Numerical experiments

In this section, we apply our Run-and-Inspect Method to a set of nonconvex problems. We admit that it is difficult to apply our theoretical results because the implicit decomposition $F = f + r$ with f, r satisfying their assumptions is not known. Nonetheless, The encouraging experimental results below demonstrate the effectiveness of our Run-and-Inspect Method on nonconvex problems even though they may not have the decomposition.

3.1 Test example : Ackley's function

The Ackley function is widely used for testing optimization algorithms, and in $\mathbb{R}^2$, it has the form

$$f(x, y) = -20e^{-0.2\sqrt{0.5(x^2+y^2)}} - e^{0.5(\cos 2\pi x + \cos 2\pi y)} + e + 20.$$

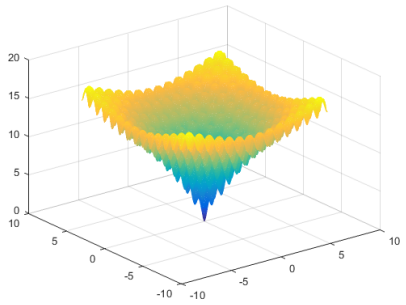

Fig. 2 Landscape of Ackley's function in $\mathbb{R}^2$.

The function is symmetric, and its oscillation is regular. To make it less peculiar, we modify it to an asymmetric function:

$$F(x, y) = -20e^{-0.04(x^2+y^2)} - e^{0.7(\sin(xy)+\sin y)+0.2\sin(x^2)} + 20. \tag{15}$$

The function F in (15) has a lot of local minimizers, which are irregularly distributed. If we simply use the gradient descent (GD) method without a good initial guess, it will converge to a nearby local minimizer. To escape from local minimizers, we conduct our Run-and-Inspect Method according to Algorithms 1 and 2. We sample points starting from the outer of the ball toward the inner. The radius R is

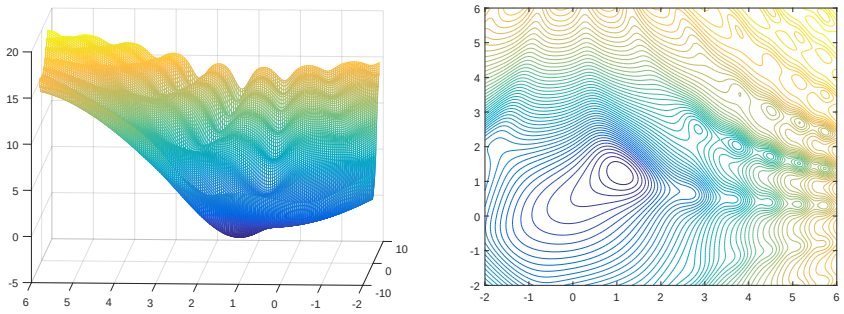

Fig. 3 Landscape and contour of F in (15).

set as 1 and ΔR as 0.2. **Alg** is GD and block-coordinate descent (BCD), and we apply two-dimensional inspection and blockwise one-dimensional inspection to them, respectively. The step size of GD and BCD is 1/40. The results are shown in Figures 4 and 5, respectively. Note that the "run" and "inspect" phases can be decoupled, so a blockwise inspection can be used with either standard descent or blockwise descent algorithms.

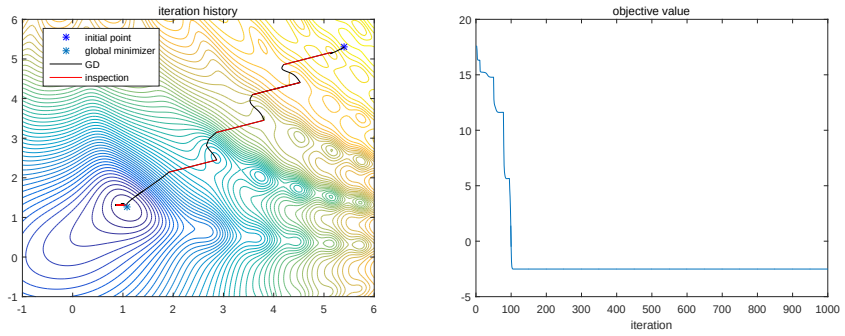

Fig. 4 GD iteration with 2D inspection

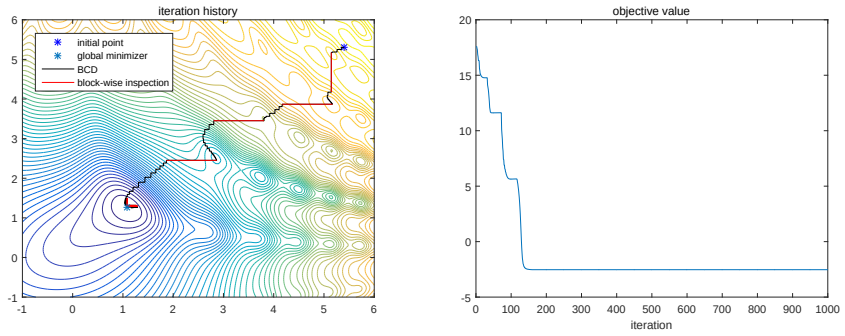

Fig. 5 BCD iteration with blockwise 1D inspection

From the figures, we can observe that blockwise inspection, which is much cheaper than standard inspection, is good at jumping out the valleys of local minimizers. Also, the inspection usually succeeds very quickly at the large initial value of R, so it is swift. These observations guide our design of inspection.

Although smaller values of R are sufficient to escape from local minimizers, especially those that are far away from the global minimizer, we empirically use a rather large R and, to limit the number of sampled points, a relatively large ΔR as well.

When an iterate is already (near) a global minimizer, there is no better point for inspection to find, so the final inspection will go through all sample points in $B(\bar{\mathbf{x}}, R)$, taking very long to complete, unlike the rapid early inspections. In most applications, however, this seems unnecessary. If F is smooth and strongly convex near the global minimizer $\mathbf{x}^*$, we can theoretically eliminate spurious local minimizers in $B(\bar{\mathbf{x}}, R')$ and thus search only in the smaller region $B(\bar{\mathbf{x}}, R) \setminus B(\bar{\mathbf{x}}, R')$. Because the function r can be nonsmooth in our assumption, we do not have $R' > 0$. But, our future work will explore more types of r. It is also worth mentioning that, in some applications, global minimizers can be recognized, for example, based on they having the desired structures, achieving the minimal objective values, or attaining certain lower bounds. If so, the final inspection can be completely avoided.

3.2 K-means clustering

Consider applying k-means clustering to a set of data $\{x_i\}_{i=1}^n \subset \mathbb{R}^d$. We assume there are K clusters $\{z_i\}_{i=1}^K$ and have the variables $\mathbf{z} = [z_1, ...z_K] \in \mathbb{R}^{d\times K}$. The problem to solve is

$$\min_{\mathbf{z}\in\mathbb{R}^{d\times K}} f(z) = \frac{1}{2n}\sum_{i=1}^{n} \min_{1\le j\le K} \|x_i - z_j\|^2.$$

A classical algorithm is the Expectation Minimization (EM) method, but it is susceptible to local minimizers. We add inspections to EM to improve its results.

We test the problems in [27]. The first problem has synthetic Guassian data in $\mathbb{R}^2$. A total of 4000 synthetic data points are generated according to four multivariate Gaussian distributions with 1000 points on each, so there are four clusters. Their means and covariance matrices are:

$$\mu_1 = \begin{bmatrix}-5\\-3\end{bmatrix},\ \mu_2 = \begin{bmatrix}5\\-3\end{bmatrix},\ \mu_3 = \begin{bmatrix}0\\5\end{bmatrix},\ \mu_4 = \begin{bmatrix}2.5\\4\end{bmatrix};$$
$$\Sigma_1 = \begin{bmatrix}0.8 & 0.1\\0.1 & 0.8\end{bmatrix},\ \Sigma_2 = \begin{bmatrix}1.2 & 0.6\\0.6 & 0.7\end{bmatrix},\ \Sigma_3 = \begin{bmatrix}0.5 & 0.05\\0.05 & 1.6\end{bmatrix},\ \Sigma_4 = \begin{bmatrix}1.5 & 0.05\\0.05 & 0.6\end{bmatrix}.$$

The EM algorithm is an iteration that alternates between labeling each data point (by associating it to the nearest cluster center) and adjusting the locations of the centers. When the labels stop updating, we start an inspection. In the above problem, the dimension of z_i is two, and we apply a 2D inspection on z_i one after one with radius $R = 10$, step size $\Delta R = 2$, and angle step size $\Delta\theta = \pi/10$. The descent threshold is $\nu = 0.1$.

The results are presented in Figure 6. We can see that the EM algorithm stops at a local minimizer but, with the help of inspection, it escapes from the local minimizer and reaches the global minimizer. This escape occurs at the first sample point in the 3rd block at radius 10 and angle $\theta = 7\pi/10$. Since the inspection succeeds on the perimeter of the search ball, it is rapid.

We also consider the Iris dataset[1], which contains 150 4-D data samples from 3 clusters. We compare the performance of the EM algorithm with and without inspection over 500 runs with their initial centers randomly selected from the data samples. We inspect the 4-D variables one after one. Rather than sampling the 4-D polar coordinates, which needs three angular axes, we only inspect two dimensional balls. That is, for center i_0 and radius r, the inspections sample the following points z_{i_0} that has only two angular variables θ_1, θ_2:

$$z_{i_0}^{\text{inspected}} = z_{i_0} + r \times [\cos\theta_1 \ \ \sin\theta_1 \ \ \cos\theta_2 \ \ \sin\theta_2]^T.$$

Such inspections are very cheap yet still effective. Similar lower-dimensional inspections should be used with high dimensional problems. We choose $R = 3$, $\Delta R = 1$, $\Delta\theta_1 = \Delta\theta_2 = \pi/10$, and a descent threshold $\nu = 10^{-3}$. The results are shown in Figure 7 and 8.

[1] https://archive.ics.uci.edu/ml/datasets/iris

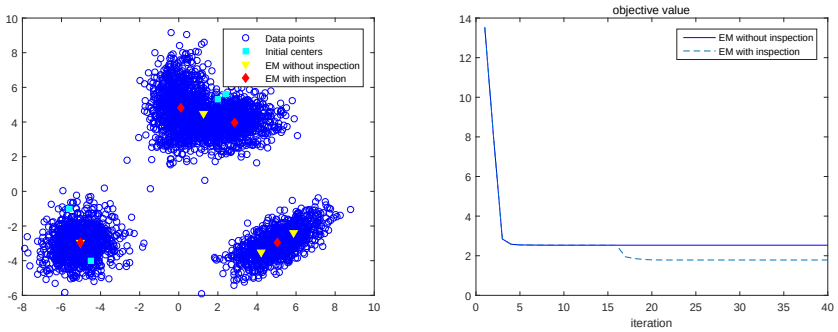

Fig. 6 Synthetic Guassian data with 4 clusters. Left: clustering result; Right: objective value in the iteration

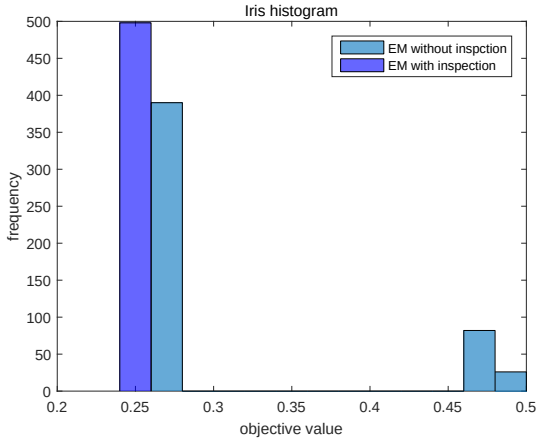

Fig. 7 histogram of the final objective values in the 500 experiments

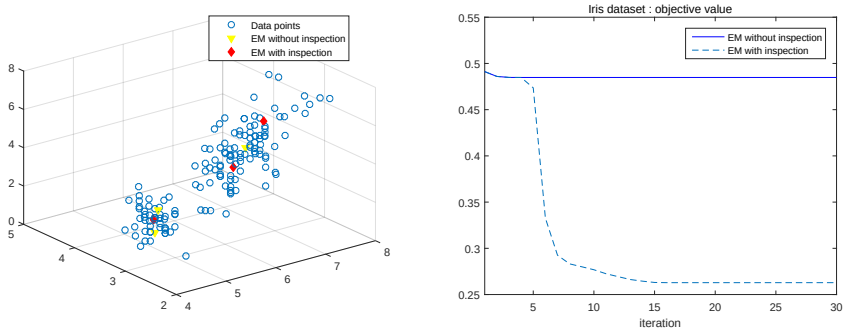

Fig. 8 left: 3-D distribution of Iris data and clustering result in one trial; right: objective value in the iteration of this trial.

Among the 500 runs, EM gets stuck at a high objective value 0.48 for 109 times. With the help of inspection, it manages to locate the optimal objective value around 0.263 every time. The average radius-at-escape during the inspections is 2, and the average number of inspections is merely 1.

3.3 Nonconvex robust linear regression

In linear regression, we are given a linear model

$$y = \langle \beta, \mathbf{x} \rangle + \varepsilon,$$

and the data points $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)$, $y_i \in \mathbb{R}, \mathbf{x}_i \in \mathbb{R}^n$. When there are outliers in the data, robustness is necessary for the regression model. Here we consider Tukey's bisquare loss, which is bounded, nonconvex and defined as:

$$\rho(r) = \begin{cases} \frac{r_0^2}{6}\{1 - (1 - (r/r_0)^2)^3\}, \text{ if } |r| < r_0, \\ \\ \frac{r_0^2}{6}, \qquad\qquad\qquad\qquad \text{otherwise.} \end{cases}$$

The empirical loss function based on ρ is

$$l(\beta) = \frac{1}{n}\sum_{i=1}^{n} \rho(y_i - \langle \beta, x_i \rangle).$$

A commonly used algorithm for this problem is the Iteratively Reweighted Least Squares (IRLS) algorithm [5], which may get stuck at a local minimizer. Our Run-and-Inspect Method can help IRLS escape from local minimizers and converge to a global minimizer. Our test uses the model

$$y = 5 + x + \varepsilon,$$

where ε is noise. We generate $x_i \sim \mathcal{N}(0, 1)$, $\varepsilon_i \sim \mathcal{N}(0, 0.5), i = 1, 2, \ldots, 20$. We also create 20% outliers by adding extra noise generated from $\mathcal{N}(0, 5)$. And we use Algorithm 1 with $R = 5, dR = 0.5, \nu = 10^{-3}$. For Tukey's function, r_0 is set to be 4.685. The results are shown in Figure 9.

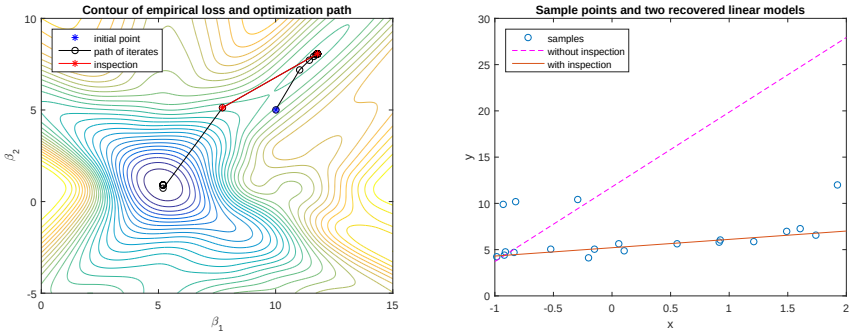

Fig. 9 The left picture displays the contour of the empirical loss $l(\beta)$ and the path of iterates. Starting from the initial point, IRLS converges to a shallow local minimum. With the help of inspection, it escapes and then converges to the global minimum. The right picture shows linear model obtained by IRLS with (red) and without (magenta) inspection.

3.4 Nonconvex compressed sensing

Given a matrix $A \in \mathbb{R}^{m \times n}$ $(m < n)$ and a sparse signal $\mathbf{x} \in \mathbb{R}^n$, we observe a vector

$$\mathbf{b} = A\mathbf{x}.$$

The problem of compressed sensing aims to recover x approximately. Besides ℓ_0 and ℓ_1-norm, $\ell_p (0 < p < 1)$ quasi-norm is often used to induce sparse solutions. Below we use $\ell_{\frac{1}{2}}$ and try to solve the problem

$$\min_{\mathbf{x} \in \mathbb{R}^n} \; Q(\mathbf{x}) := \frac{1}{2}\|A\mathbf{x} - \mathbf{b}\|^2 + \lambda \|\mathbf{x}\|_{\frac{1}{2}}^{\frac{1}{2}},$$

by cyclic coordinate update. At iteration k, it updates the jth coordinate, where $j = \mathrm{mod}(k, n) + 1$, via

$$x_j^{k+1} = \underset{x_j}{\mathrm{argmin}}\, Q(x_j, \mathbf{x}_{-j}^k) \tag{16}$$

$$= \underset{x_j}{\mathrm{argmin}}\ \frac{1}{2}A_j^T A_j x_j^2 + A_j^T(A\mathbf{x}^k - \mathbf{b})x_j + \lambda\sqrt{|x_j|}. \tag{17}$$

It has been proved in [26] that (16) has a closed-form solution. Define

$$B_{j,\mu}(\mathbf{x}) = x_j - \mu A_j^T(A\mathbf{x} - \mathbf{b}),$$

$$H_{\lambda,\frac{1}{2}}(z) = \begin{cases} \frac{2}{3}z(1+\cos(\frac{2\pi}{3} - \frac{2}{3}\arccos(\frac{\lambda}{4}(\frac{|z|}{3})^{-\frac{3}{2}}))), & \text{if } |z| > \frac{\sqrt[3]{54}}{4}(2\lambda)^{\frac{2}{3}}, \\ 0, & \text{otherwise.} \end{cases}$$

Then

$$x_j^{k+1} = H_{\lambda\mu,1/2}(B_{j,\mu}(\mathbf{x}^k)),$$

where $\mu = \|A_j\|^2$. In our experiments, we choose $m = 25, 50, 100$ and $n = 2m$. The elements of A are generated from $\mathcal{U}(0, \frac{1}{\sqrt{m}})$ i.i.d. The vector $\mathbf{x}$ has 10% nonzeros with their values generated from $\mathcal{U}(0.2, 0.8)$ i.i.d. Set $b = A\mathbf{x}$. Here, we apply coordinate descent with inspection (CDI), and compared it with standard coordinate descent (CD) and half thresholding algorithm (*half*) [26]. For every pair of (m, n), we choose the parameter $\lambda = 0.05$ and run 100 experiments. When the iterates stagnate at a local minimizer $\bar{\mathbf{x}}$, we perform a blockwise inspection with each block consisting of two coordinates. Checking all pairs of two coordinates is expensive and not necessary since $\bar{\mathbf{x}}$ is sparse. We improve the efficiency by pairing only i, j where $x_i \neq 0, x_j = 0$. Similar to previous experiments, we sample points from the outer of the 2D ball toward the inner. We choose $R = 0.5$, $\Delta R = 0.05$. The results are presented in Table 1 and Figure 10. CDI shows a significant improvement over its competitors.

n, p	algorithm	a	b	c	ave obj
$n = 25$	*half*	47.73%	2	2	0.0365
$p = 50$	CD	62.40%	25	27	0.0272
	CDI	83.95%	65	69	0.0208
$n = 50$	*half*	46.43%	0	0	0.0736
$p = 100$	CD	76.39%	24	32	0.0443
	CDI	92.34%	57	68	0.0369
$n = 100$	*half*	44.31%	0	0	0.1622
$p = 200$	CD	85.97%	10	18	0.0795
	CDI	94.31%	54	76	0.0756

Table 1 Statistics of 100 compressed sensing problems solved by three $\ell_{\frac{1}{2}}$ algorithms

1. *half*: iterative half thresholding; CD: coordinate descent; CDI: CD with inspection.
2. a is the ratio of correctly identified nonzeros to true nonzeros, averaged over the 100 tests (100% is impossible due to noise and model error); b is the number of tests with all true nonzeros identified; c is the number of tests in which the returned points yield lower objective values than that of the true signal (only model error, no algorithm error). Higher a, b, c are better.
3. "ave obj" is the average of the objective values; lower is better.

3.5 Nonconvex Sparse Logistic Regression

Logistic regression is a widely-used model for classification. Usually we are given a set of training data $\{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^N$, where $\mathbf{x}^{(i)} \in \mathbb{R}^d$ and $y^{(i)} \in \{0, 1\}$. The label y is assumed to satisfy the following conditional distribution:

$$\begin{cases} p(y = 1|\mathbf{x}; \theta) = \frac{1}{1+\exp(-\theta^T\mathbf{x})}, \\ p(y = 0|\mathbf{x}; \theta) = \frac{1}{1+\exp(\theta^T\mathbf{x})}, \end{cases} \tag{18}$$

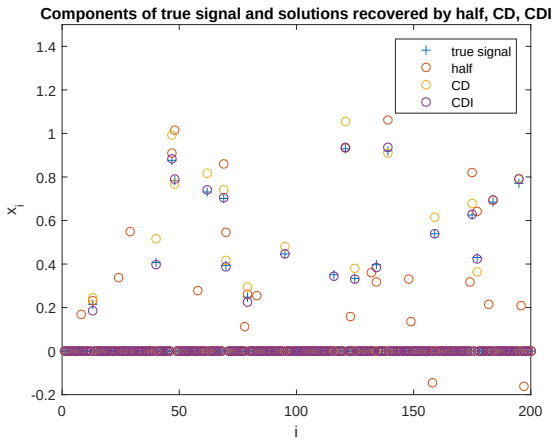

Fig. 10 Comparison of the true signal x and signals recovered from *half*, CD, CDI.

In one experiment, CDI recovered all positions of nonzeros of x, while CD failed to recover x_{116}, x_{134}. The *half* algorithm just got stuck at a local minizer far from x.

where θ is the model parameter.

To learn θ, we minimize the negative log-likelihood function

$$l(\theta) = \sum_{i=1}^{N} -\log p(y^{(i)}|\mathbf{x}^{(i)};\theta),$$

which is convex and differentiable. When N is relatively small, we need variable selection to avoid overfitting. In this test, we use the minimax concave penalty (MCP) [28]:

$$p^{\mathrm{MCP}}_{\lambda,\gamma}(x) = \begin{cases} \lambda|x| - \frac{x^2}{2\gamma}, & \text{if } |x| \le \gamma\lambda, \\ \frac{1}{2}\gamma\lambda^2, & \text{if } |x| > \gamma\lambda. \end{cases}$$

The θ-recovery model writes

$$\min_{\theta} l(\theta) + \beta p^{\mathrm{MCP}}_{\lambda,\gamma}(\theta).$$

The penalty $p^{\mathrm{MCP}}_{\lambda,\gamma}$ is proximable with

$$\mathrm{prox}_p(z) = \begin{cases} \frac{\gamma}{\gamma-1} S_\lambda(z) & \text{if } |z| \le \gamma\lambda, \\ z & \text{if } |z| > \gamma\lambda \end{cases}$$

where $S_\lambda(z) = (|z| - \lambda)_+ \,\mathrm{sign}(z)$.

We apply the prox-linear (PL) algorithm to solve this problem. When it nearly converges, inspection is then applied. We design our experiments according to [20]: we consider $d = 50$ and $N = 200$ and assume the true θ^* has K non-zero entries. In the training procedure, we generate data from i.i.d. standard Gaussian distribution, and we randomly choose K non-zero elements with i.i.d standard Gaussian distribution to form θ^*. The labels are generated by $y = 1(\mathbf{x}^T\theta + w \ge 0)$, where w is sampled according to the Gaussian distribution $\mathcal{N}(0, \epsilon^2 I)$. We use PL iteration with and without inspection to recover θ. After that, we generate 1000 random test data points to compute the test error of the θ. We set the parameter $\beta = 1.5 - 0.06 \times K$, $\lambda = 1$, $\gamma = 5$ and the step size 0.5 for PL iteration. For each K and ϵ, we run 100 experiments and calculate the mean and variance of the results. The inspection parameters are $R = 5$, $\Delta R = 1$, and $\Delta\theta = \pi/10$. The sample points in inspections are similar to those in section 3.4. The results are presented in Table 2. The objective values and test errors of PLI, the PL algorithm with inspection, are significantly better than the native PL algorithm. On the other hand, the cost is also 3 – 6 times as high.

We plot the convergence history of the objective values in one trial and the recovered θ in Figure 11. It is clear that the inspection works in learning a better θ by reaching a smaller objective value.

K, ϵ	algorithm	average #.iterations / #.inspections	objective value		test error	
			mean	var	mean	var
$K = 5$	PL	594	48.0	305	7.26%	1.55e-03
$\epsilon = 0.01$	PLI	3430/11.43	26.8	44.9	3.79%	5.43e-04
$K = 5$	PL	601	52.7	409	7.81%	1.29e-03
$\epsilon = 0.1$	PLI	2280/7.98	33.8	51.9	4.38%	5.43e-04
$K = 10$	PL	1040	43.6	87.0	8.42%	8.68e-04
$\epsilon = 0.01$	PLI	2610/4.78	33.5	35.9	5.73%	5.73e-04
$K = 10$	PL	990	47.5	87.2	9.41%	9.88e-04
$\epsilon = 0.1$	PLI	2370/3.86	36.3	40.1	5.69%	5.50e-04
$K = 15$	PL	1600	36.2	54.3	7.85%	8.29e-04
$\epsilon = 0.01$	PLI	3010/3.21	29.2	17.5	5.77%	5.20e-04
$K = 15$	PL	1570	37.1	40.2	7.80%	8.30e-04
$\epsilon = 0.1$	PLI	2820/2.77	30.7	16.0	6.66%	4.63e-04

Table 2 Sparse logistic regression results of 100 tests. PL is the prox-linear algorithm. PLI is the PL algorithm with inspection. "var" is variance.

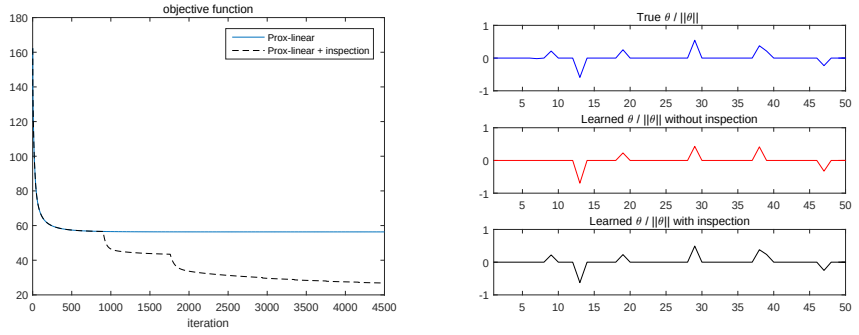

Fig. 11 Sparse logistic regression result in one trial.

4 Conclusions

In this paper, we have proposed a simple and efficient method for nonconvex optimization, based on our analysis of R-local minimizers. The method applies local inspections to escape from local minimizers or verify the current point is an R-local minimizer. For a function that can be implicitly decomposed to a smooth, strongly convex function plus a restricted nonconvex functions, our method returns an (approximate) global minimizer. Although some of the tested problems may not possess the assumed decomposition, numerical experiments support the effectiveness of the proposed method.

References

1. Chaudhari, P., Choromanska, A., Soatto, S., LeCun, Y.: Entropy-SGD: Biasing gradient descent into wide valleys. arXiv preprint arXiv:1611.01838 (2016)
2. Chaudhari, P., Oberman, A., Osher, S., Soatto, S., Carlier, G.: Deep relaxation: Partial differential equations for optimizing deep neural networks. arXiv preprint arXiv:1704.04932 (2017)
3. Conn, A.R., Gould, N.I., Toint, P.L.: Trust region methods. SIAM (2000)
4. Conn, A.R., Scheinberg, K., Vicente, L.N.: Introduction to Derivative-Free Optimization. No. 8 in MPS-SIAM series on optimization. Society for Industrial and Applied Mathematics / Mathematical Programming Society, Philadelphia (2009)
5. Fox, J.: An R and S-Plus Companion to Applied Regression. Sage Publications, Thousand Oaks, Calif (2002)
6. Ge, R., Huang, F., Jin, C., Yuan, Y.: Escaping from saddle points—online stochastic gradient for tensor decomposition. In: Conference on Learning Theory, pp. 797–842 (2015)
7. Ge, R., Lee, J.D., Ma, T.: Matrix completion has no spurious local minimum. In: Advances in Neural Information Processing Systems, pp. 2973–2981 (2016)
8. Ghadimi, S., Lan, G.: Accelerated gradient methods for nonconvex nonlinear and stochastic programming. Mathematical Programming **156**(1-2), 59–99 (2016)
9. Jin, C., Ge, R., Netrapalli, P., Kakade, S.M., Jordan, M.I.: How to escape saddle points efficiently. arXiv preprint arXiv:1703.00887 (2017)

10. Karimi, H., Nutini, J., Schmidt, M.: Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition. arXiv:1608.04636 (2016)
11. Kirkpatrick, S., Gelatt Jr, C.D., Vecchi, M.P.: Optimization by simulated annealing. In: Spin Glass Theory and Beyond: An Introduction to the Replica Method and Its Applications, pp. 339–348. World Scientific (1987)
12. Martínez, J.M., Raydan, M.: Cubic-regularization counterpart of a variable-norm trust-region method for unconstrained minimization. Journal of Global Optimization **68**(2), 367–385 (2017)
13. Nesterov, Y., Polyak, B.T.: Cubic regularization of Newton method and its global performance. Mathematical Programming **108**(1), 177–205 (2006)
14. Panageas, I., Piliouras, G.: Gradient descent converges to minimizers: The case of non-isolated critical points. CoRR, abs/1605.00405 (2016)
15. Pascanu, R., Dauphin, Y.N., Ganguli, S., Bengio, Y.: On the saddle point problem for non-convex optimization. arXiv preprint arXiv:1405.4604 (2014)
16. Peng, Z., Wu, T., Xu, Y., Yan, M., Yin, W.: Coordinate friendly structures, algorithms and applications. Annals of Mathematical Sciences and Applications **1**(1), 57–119 (2016)
17. Polyak, B.T.: Gradient methods for minimizing functionals. Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki **3**(4), 643–653 (1963)
18. Reddi, S.J., Hefny, A., Sra, S., Poczos, B., Smola, A.: Stochastic variance reduction for nonconvex optimization. In: International conference on machine learning, pp. 314–323 (2016)
19. Sagun, L., Bottou, L., LeCun, Y.: Singularity of the Hessian in deep learning. arXiv preprint arXiv:1611.07476 (2016)
20. Shen, X., Gu, Y.: Nonconvex sparse logistic regression with weakly convex regularization. arXiv preprint arXiv:1708.02059 (2017)
21. Sun, J., Qu, Q., Wright, J.: Complete dictionary recovery over the sphere. In: Sampling Theory and Applications (SampTA), 2015 International Conference on, pp. 407–410. IEEE (2015)
22. Sun, J., Qu, Q., Wright, J.: A geometric analysis of phase retrieval. In: Information Theory (ISIT), 2016 IEEE International Symposium on, pp. 2379–2383. IEEE (2016)
23. Wang, Y., Yin, W., Zeng, J.: Global convergence of admm in nonconvex nonsmooth optimization. arXiv preprint arXiv:1511.06324 (2015)
24. Wu, L., Zhu, Z., Weinan, E.: Towards understanding generalization of deep learning: Perspective of loss landscapes. arXiv preprint arXiv:1706.10239 (2017)
25. Xu, Y., Yin, W.: A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. SIAM Journal on imaging sciences **6**(3), 1758–1789 (2013)
26. Xu, Z., Chang, X., Xu, F., Zhang, H.: $l_{1/2}$ regularization: A thresholding representation theory and a fast solver. IEEE Transactions on neural networks and learning systems **23**(7), 1013–1027 (2012)
27. Yin, P., Pham, M., Oberman, A., Osher, S.: Stochastic backward Euler: An implicit gradient descent algorithm for k-means clustering. arXiv preprint arXiv:1710.07746 (2017)
28. Zhang, C.H.: Nearly unbiased variable selection under minimax concave penalty. The Annals of Statistics **38**(2), 894–942 (2010)