Gradient Descent Can Take Exponential Time to Escape Saddle Points

NIPS 2017 (spotlight)

Simon S. Du, Chi Jin, Michael Jordan et al.

CMU, UCB & USC

Closely Related Work

- Ge, Rong, et al. "**Escaping from saddle points—online stochastic gradient for tensor decomposition**." *COLT*. 2015.
- Lee, Jason D., et al. "**Gradient descent only converges to minimizers**." *COLT*. 2016.
- Kawaguchi, Kenji. "**Deep learning without poor local minima**." *NIPS. 2016.*
- Ge, Rong, Chi Jin, and Yi Zheng. "**No Spurious Local Minima in Nonconvex Low Rank Problems: A Unified Geometric Analysis**." *ICML. 2017.*
- Jin, Chi, et al. "**How to Escape Saddle Points Efficiently**." *ICML. 2017.*
- Gonen, Alon, and Shai Shalev-Shwartz. "**Fast Rates for Empirical Risk Minimization of Strict Saddle Problems**." *COLT. 2017.*

General Optimization Problem

- Problem

$$\min f(x)$$

$$x \in S, S \subseteq R^n$$

- A common solution: Gradient Descent (GD)

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

$\eta > 0$ is a learning rate

$\nabla f(x_k)$ is the gradient at x_k

Assumption: Existence of gradient

Theoretical Guarantee of GD

- Stationary point (critical point)

Local minimizer

Local maximizer

Saddle point

$$\nabla f(x^*) = 0, \forall x^* \in S$$

- Guarantee of GD

$$\nabla f(x_K) \leq \epsilon, \text{with } \epsilon > 0$$

$K \leq O(poly(\epsilon))$ is the number of iterations

Nesterov, Yurii. *Introductory lectures on convex optimization: A basic course*. 2004.

Taxonomy

- Convex optimization: critical point ⇔ globally optimal

Condition	Time complexity	Acceleration
Convex and deterministic	$K = O\left(\frac{1}{\epsilon}\right)$	$K = O\left(\frac{1}{\epsilon^{0.5}}\right)$
Convex and stochastic	$K = O\left(\frac{1}{\epsilon^2}\right)$	$K = O\left(\frac{1}{\epsilon}\log(\frac{1}{\epsilon})\right)$
Convex and adversarial	$K = O\left(\frac{1}{\epsilon^2}\right)$	No result

- Non-convex optimization: critical point { Local minimizer / Saddle point }

Condition	Time complexity
Convex and deterministic	polynomial time
Convex and stochastic	No result

Non-convex: Critical Point ⇔ Minimizer?

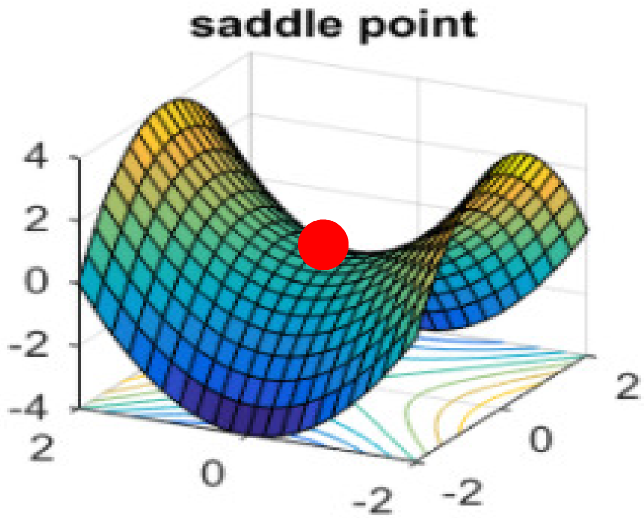
- Can we escape saddle points via GD? YES

  Lee, Jason D., et al. "**Gradient descent only converges to minimizers**." *COLT*. 2016.

- What is the time complexity of the escaping?
  - Can take exponential time (✓)
  - Can take polynomial time

Definition of Saddle Points

- A strict saddle point x^*
  - There exists a $\alpha > 0$, such that $||\nabla f(x^*)||_2 = 0$ and $\lambda_{\min}\left(\nabla^2 f(x^*)\right) \leq -\alpha$.
  - The minimal eigenvalue of Hessian matrix is strictly negative



http://www.offconvex.org/2016/03/22/saddlepoints/

Saddle Point in $f(x_1, x_2) = x_1^2 - x_2^2$

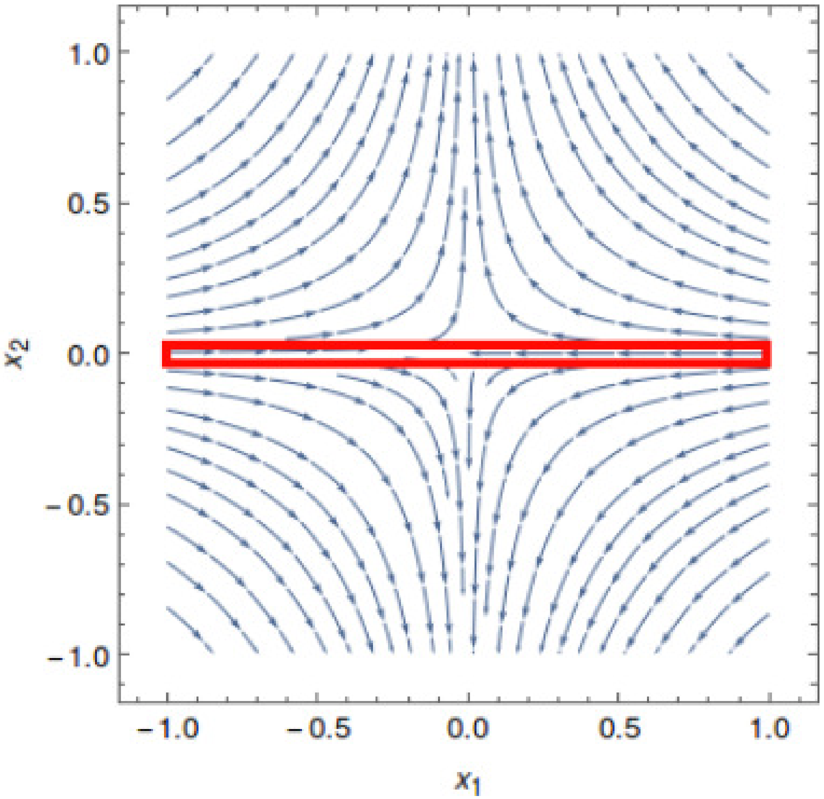
- A saddle point is $(0,0)$
- Given $\eta = \frac{1}{4}$, the update rules are

$$x_1^{k+1} = \frac{x_1^k}{2} \qquad x_2^{k+1} = \frac{3x_2^k}{2}$$

- Consider initialization in the region as

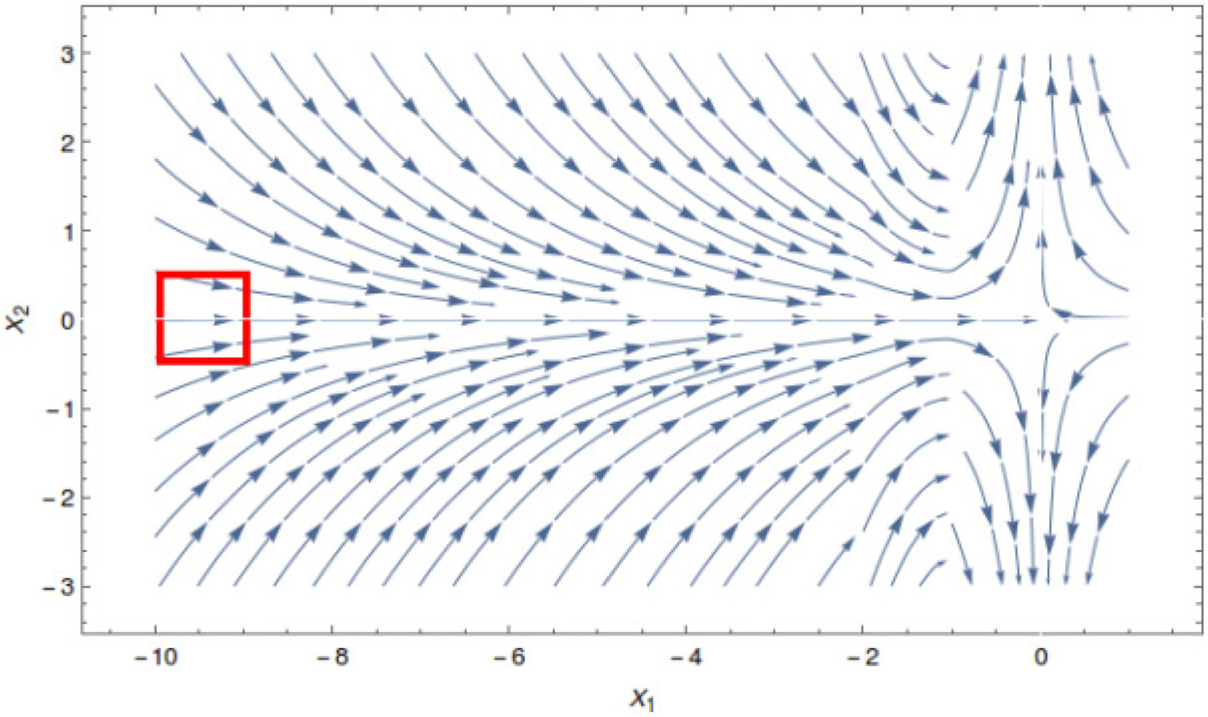
$[-1,1] \times \left[-\left(\frac{3}{2}\right)^{-\exp\left(\frac{1}{\epsilon}\right)}, \left(\frac{3}{2}\right)^{-\exp\left(\frac{1}{\epsilon}\right)}\right]$, the updating step is exponential.

Demonstration of Gradient Field



$$f(x_1, x_2) = x_1^2 - x_2^2$$

Another Example



Exponentially far away

Exponential Time Complexity

- Two examples to show exponential time complexity with a specific initialization
- How about some random initializations?

Theorem 4.1 (Uniform initialization over a unit cube). *Suppose the initialization point is uniformly sampled from* $[-1, 1]^d$. *There exists a function* f *defined on* $\mathbb{R}^d$ *that is* B*-bounded,* ℓ*-gradient Lipschitz and* ρ*-Hessian Lipschitz with parameters* B, ℓ, ρ *at most* $\text{poly}(d)$ *such that:*

1. *with probability one, gradient descent with step size* $\eta \leq 1/\ell$ *will be* $\Omega(1)$ *distance away from any local minima for any* $T \leq e^{\Omega(d)}$.
2. *for any* $\epsilon > 0$*, with probability* $1 - e^{-d}$*, perturbed gradient descent (Algorithm 1) will find a point* x *such that* $\|x - x^*\|_2 \leq \epsilon$ *for some local minimum* x^* *in* $\text{poly}(d, \frac{1}{\epsilon})$ *iterations.*

*Jin, Chi, et al. "**How to Escape Saddle Points Efficiently**." ICML. 2017.*

Proof Sketch

- Construct a function with 2^d symmetric minima
- The saddle points are of the form

$$(\pm c, \cdots, \pm c, 0, \cdots, 0)$$

- Then GD will travel across d neighborhoods of saddle points
- Prove the number of iterations to escape each saddle point should be κ^i with $i \in \{1, \cdots, d\}$
- Thus the total time complexity is exponential

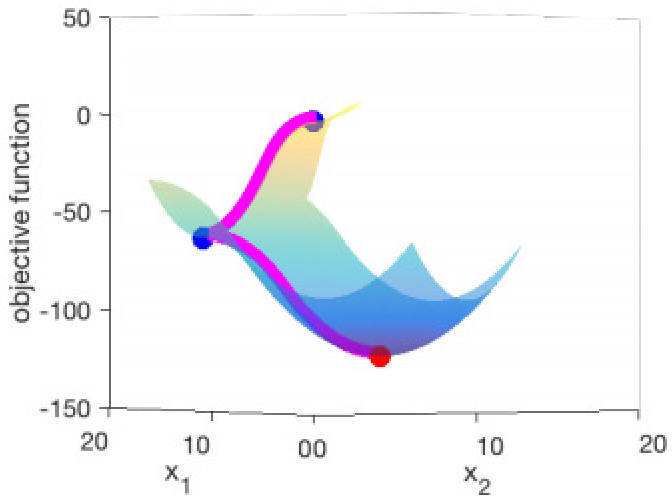
Discussions of The Paper

- Conclusion
  - GD can encounter non-convex functions leading to exponential steps to escape the saddle points
- Two interesting questions
  - What kind of non-convex functions that GD can take polynomial steps to escape the saddle points?
  - Does the stochastic GD have the same property?

  (That is, SGD can be exponential in time complexity to escape the saddle points.)

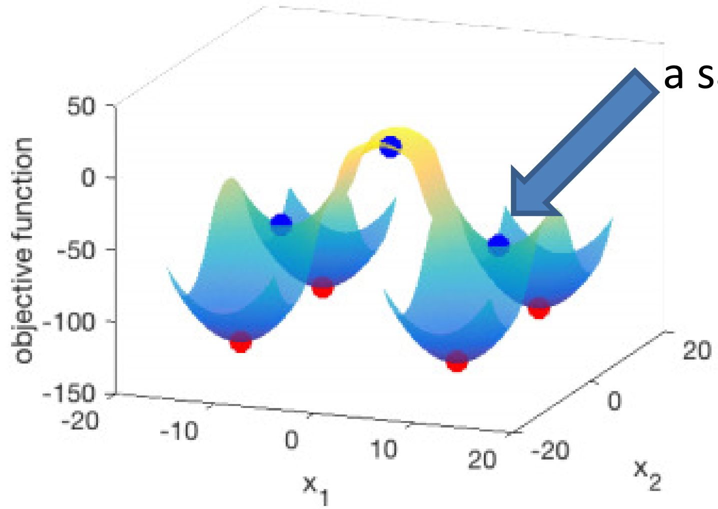
Why Escaping Saddle Points?

- Convex optimization
  - Every local minimizer is global (local-global rule)
- Non-convex optimization
  - Generally, it is NP-hard and has no local-global rule



Escaping Saddle Points to Be Globally Optimal

- Tensor decomposition (non-convex)
  - Local minimal point is global optimal in the fourth order tensor decomposition



Ge, Rong, et al. "**Escaping from saddle points—online stochastic gradient for tensor decomposition**." *COLT*. 2015.

Escaping Saddle Points to Be Globally Optimal

- Non-convex low rank problem
  - All local minima are also globally optimal
  - No high-order saddle points exist

Ge, Rong, Chi Jin, and Yi Zheng. "**No Spurious Local Minima in Nonconvex Low Rank Problems: A Unified Geometric Analysis**." *ICML. 2017.*

- Deep learning with feedforward neural networks
  - For any deep neural network, any local minimum is global and also escaping the saddle points is guaranteed to obtain a globally minimum point.
  - Model: $Y(W, X) = W_h \times W_{h-1} \times W_1 \times X$

Kawaguchi, Kenji. "**Deep learning without poor local minima**." *NIPS. 2016.*

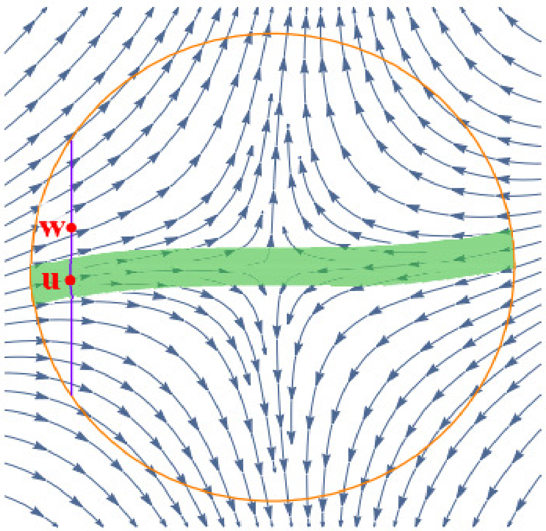
How To Escape Saddle Points?

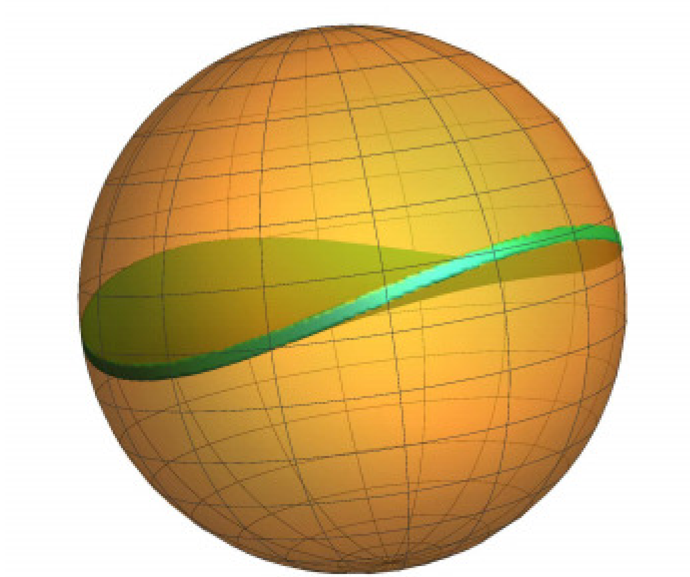
- Perturbation

Algorithm 1 Perturbed Gradient Descent (Meta-algorithm)

---

for $t = 0, 1, \ldots$ **do**
  if perturbation condition holds **then**
    $\mathbf{x}_t \leftarrow \mathbf{x}_t + \xi_t, \qquad \xi_t$ uniformly $\sim \mathbb{B}_0(r)$
  $\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t - \eta \nabla f(\mathbf{x}_t)$





Jin, Chi, et al. "**How to Escape Saddle Points Efficiently**." *ICML. 2017.*

Final Discussions

- Remarks
  - Escaping saddle points is important in non-convex optimization
  - Perturbation gradient descent (PGD) powers the solution in non-convex optimization
- Questions
  - What is the optimal order of PGD in non-convex optimization?
  - What kind of noises helps escaping saddle points?
  - Does the adding noise depend on the learning data?

Gonen, Alon, and Shai Shalev-Shwartz. "**Fast Rates for Empirical Risk Minimization of Strict Saddle Problems**." *COLT. 2017.*