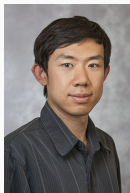


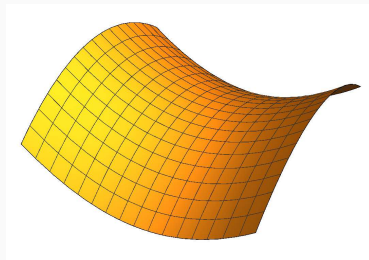
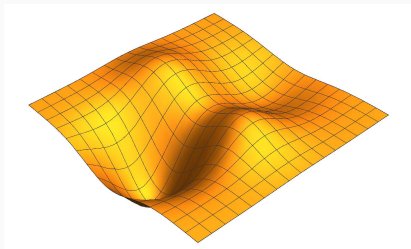
On the Local Minima of the Empirical Risk

Chi Jin^{*1}, Lydia T. Liu^{*1}, Rong Ge², Michael I. Jordan¹

¹EECS, University of California, Berkeley. ²Duke University.

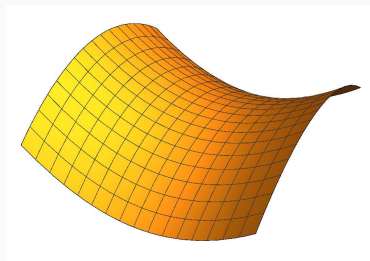
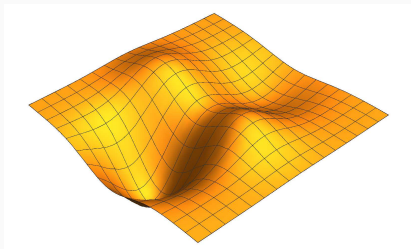


Nonconvex Optimization.



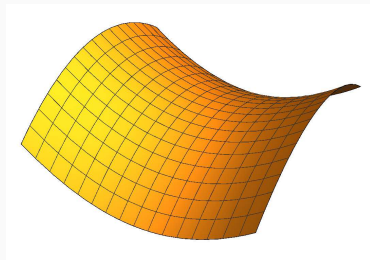
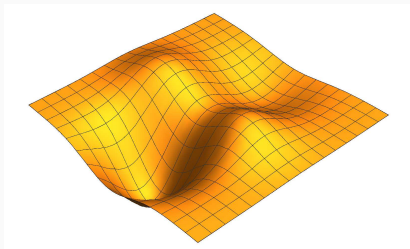
- Gradient Descent (GD) → stationary points: local max, saddle points, local min.

Nonconvex Optimization.



- ▶ Gradient Descent (GD) → **stationary points**: local max, saddle points, local min.
- ▶ Perturbed GD [Jin et al. 2017] **efficiently** escapes local max and saddle points.

Nonconvex Optimization.

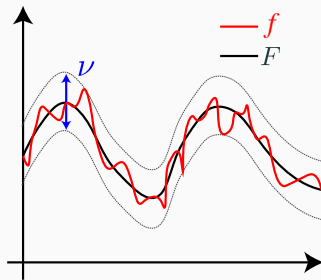


- ▶ Gradient Descent (GD) → **stationary points**: local max, saddle points, local min.
- ▶ Perturbed GD [Jin et al. 2017] **efficiently** escapes local max and saddle points.
- ▶ **How to deal with spurious local min?**

In general, finding global minima is **NP-hard**.

Local Minima

In general, finding global minima is **NP-hard**.



Avoiding “shallow” local minima

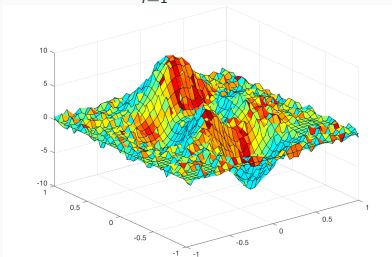
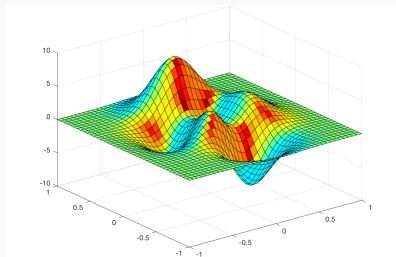
Goal: finds **approximate local minima** of **smooth** nonconvex function F , given only access to an erroneous version f where $\sup_{\mathbf{x}} |F(\mathbf{x}) - f(\mathbf{x})| \leq \nu$

Statistical Learning.

Minimize population risk R while only have access to empirical risk $\hat{R}_n$.

$$R(\theta) = \mathbb{E}_{\mathbf{z} \sim \mathcal{D}}[L(\theta; \mathbf{z})],$$

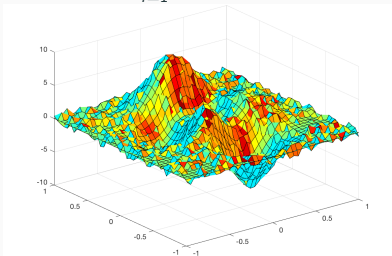
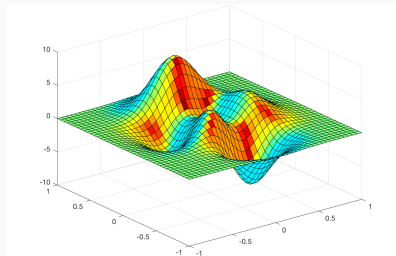
$$\hat{R}_n(\theta) = \frac{1}{n} \sum_{i=1}^n L(\theta; \mathbf{z}_i).$$



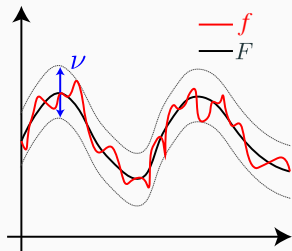
Statistical Learning.

Minimize population risk R while only have access to empirical risk $\hat{R}_n$.

$$R(\theta) = \mathbb{E}_{\mathbf{z} \sim \mathcal{D}}[L(\theta; \mathbf{z})], \quad \hat{R}_n(\theta) = \frac{1}{n} \sum_{i=1}^n L(\theta; \mathbf{z}_i).$$



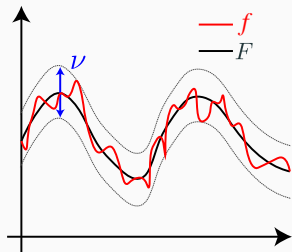
Unifrom convergence guarantees $\sup_{\theta} |R(\theta) - \hat{R}_n(\theta)| \leq O(1/\sqrt{n})$.



Goal: find ϵ -approximate local minima of F in polynomial time.

Central Questions:

1. What algorithm can achieve this?
2. How much error ν can be tolerated?

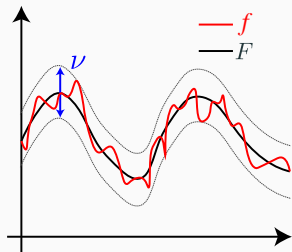


Goal: find ϵ -approximate local minima of F in polynomial time.

Central Questions:

1. What algorithm can achieve this?
2. How much error ν can be tolerated?

Zhang et al. [2017]: Stochastic Gradient Langevin Dynamics (SGLD) if $\nu \leq \epsilon^2/d^8$.



Goal: find ϵ -approximate local minima of F in polynomial time.

Central Questions:

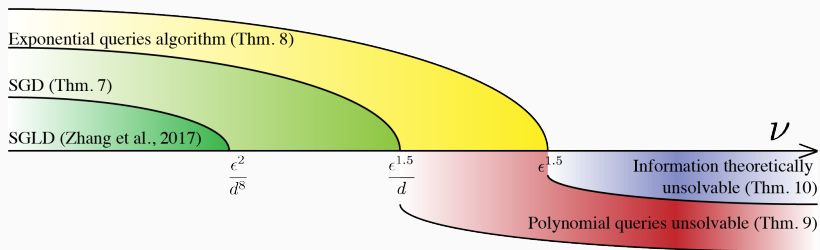
1. What algorithm can achieve this?
2. How much error ν can be tolerated?

Zhang et al. [2017]: Stochastic Gradient Langevin Dynamics (SGLD) if $\nu \leq \epsilon^2/d^8$.

This Work: Perturbed SGD on a “smoothed” version of f if $\nu \leq \epsilon^{1.5}/d$.

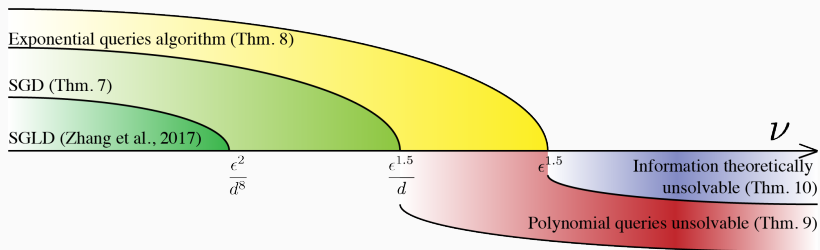
Is there better polynomial time algorithms that tolerate larger error?

Is there better polynomial time algorithms that tolerate larger error? **No!**



Complete characterization of error ν vs accuracy ϵ and dimension d .

Is there better polynomial time algorithms that tolerate larger error? **No!**



Complete characterization of error ν vs accuracy ϵ and dimension d .

Poster: Wed 5-7 PM, #43. Thanks!