Introduction	
000	
0000	

# Sample Complexity of Gradient Methods in Stochastic Convex Optimization

Genghis Luo, Nikos Tsilivis

NYU Shanghai, NYU CDS

April 8th, 2025



Introduction 000 0000	Sample complexity of GD & SGD in SCO	Conclusion O	Appendix 00

#### Outline

Introduction

- Stochastic Convex Optimization
- Failure of ERM!
- Stability & Regularized ERM
- Sample complexity of GD & SGD in SCO
  - Algorithms
  - Sample complexity of GD
  - Sample complexity of SGD

# 3 Conclusion

# References

# Appendix

#### **Stochastic Convex Optimization – Setup**

## Setup

- Instance set  $\mathcal{Z}$ : arbitrary measurable set.
- Hypothesis set W: closed, convex subset of a Hilbert space.
   For instance, W ⊆ ℝ<sup>d</sup>.
- Distribution  $\mathcal{D}$  over  $\mathcal{Z}$ .
- ► Risk/Loss function f : W × Z → R: convex and Lipschitz-continuous w.r.t. its first argument w.
- Population loss:  $F(w) = \mathbb{E}_{z \sim D}[f(w, z)].$
- Empirical loss:  $\hat{F}(w) = \frac{1}{m} \sum_{i=1}^{m} f(w, z_i)$  for  $z_1, \ldots, z_m \sim \mathcal{D}$ .

**Goal**: Find  $w^* \in \mathcal{W}$  such that population loss gets minimized:

$$w^{\star} \in \arg\min_{w \in \mathcal{W}} F(w)$$

This is an instance of the General Setting of Learning introduced by Vapnik in 1995.

Introduction ○●○ ○○○○	Sample complexity of GD & SGD in SCO	Conclusion O	Appendix 00

#### Stochastic Convex Optimization – Example

Take:

$$\ \, {\mathfrak Z}={\mathcal X}\times\{\pm1\}, \text{ where } {\mathcal X}=\{x\in {\mathbb R}^d: \|x\|_2\leq B\}.$$

If (w, (x, y)) = l((w, x), y) for some convex and Lipschitz loss function l.

**Uniform convergence**. For any  $\mathcal{D}$ , with high probability over  $z_1, \ldots, z_m \sim \mathcal{D}$ :

$$\sup_{w\in\mathcal{W}}\left|F(w)-\hat{F}(w)\right|\xrightarrow{m\to\infty}0.$$

This justifies choosing the *empirical risk minimizer* (ERM):

$$\hat{w} = \arg\min_{w} \hat{F}(w)$$

and guarantees that  $F(\hat{w})$  converges to  $F(w^*) = \inf_w F(w)$  as m increases (learnable).



#### Reminder – Supervised learning

## In Supervised Learning (e.g. classification):

► Loss 
$$f(h, (x, y)) = \mathbb{1}{h(x) \neq y}$$
 (not convex)



Learnability is equivalent to the success of ERM.

What about problems in Stochastic Convex Optimization (SCO)?



## [SSSSS10] construct an instance, where:

▶ ERM fails! With high probability over the draw of the dataset:

$$F(\hat{w}) - \hat{F}(\hat{w}) = \frac{1}{2} > 0.$$
 (1)

Uniform convergence does not hold! With high probability over the draw of the dataset:

$$\sup_{w} \left| F(w) - \hat{F}(w) \right| \geq \frac{1}{2}.$$
 (2)

Construction: In infinite dimensions.

	ntroduction	Sample complexity of GD & SGD in SCO	Conclusion O		Appen 00
--	-------------	--------------------------------------	-----------------	--	-------------

Dimension dependent sample complexity

- Related question: How large does the sample size *m* need to be with respect to the input dimension *d* to guarantee that ERM generalizes?
- Construction in finite dimensions due to [Fel16]. See board.



Figure 1: Basic construction for d = 2.

#### **Regularized ERM**

## Yet, regularized ERM always succeeds in SCO.

## Theorem 1 ([SSSSS10])

Let  $f : \mathcal{W} \times \mathcal{Z} \to \mathbb{R}$  be such that  $\mathcal{W}$  is bounded by B and f(w, z) is convex and L-Lipschitz with respect to w. Let  $z_1, \ldots, z_m$  be an *i.i.d.* sample and let  $\hat{w}_{\lambda}$  be defined as:

$$\hat{w}_{\lambda} = \arg\min_{w \in \mathcal{W}} \left( rac{1}{m} \sum_{i=1}^m f(w, z_i) + rac{\lambda}{2} \|w\|^2 
ight),$$

for  $\lambda = \sqrt{rac{16L^2}{\delta B^2 m}}$ . Then, with probability at least  $1-\delta$  we have:

$$F(\hat{w}_{\lambda}) - F(w^*) \leq \sqrt{\frac{8L^2B^2}{\delta m}}$$



Proof of previous theorem via stability. Reminder:

## Definition 2 (Uniform Stability)

Let S and S' be any two training samples that differ by a single point. A learning algorithm  $\mathcal{A}$  is said to be *uniformly*  $\beta$ -stable if the hypotheses it returns when trained on S and S' satisfy

$$\forall z \in \mathcal{Z}, \quad |f(w_S, z) - f(w_{S'}, z)| \leq \beta.$$

The smallest such  $\beta$  satisfying this inequality is called the *stability coefficient* of A.

Furthermore, [SSSSS10] show that stability is also sufficient for learnability (in the General Setting of Learning of Vapnik).

#### Learnability in SCO – Connection to OCO

- Alternative algorithm: Online projected Subgradient Descent [Zin03] and Online-to-Batch conversion.
- The output  $\tilde{w}$  of the online-to-batch algorithm is **NOT** an empirical minimizer.
- Regularization is implicit in the definition of the algorithm. See [SS07] for details.



▶ [SSSSS10] We say that a learning rule  $\mathcal{A} : \bigcup_{m=1}^{\infty} \mathcal{Z}^m \mapsto \mathcal{W}$  is an Asympotic ERM with rate  $\epsilon_{\text{ERM}}(m)$  under  $\mathcal{D}$  if:

$$\mathbb{E}_{\mathcal{S}\sim\mathcal{D}^m}\left[\hat{F}(\mathcal{A}(\mathcal{S}))-\hat{F}(\hat{w})
ight]\leq\epsilon_{ ext{ERM}}(m).$$



Remark: Regularized ERM of Theorem 1 is an Asymptotic ERM.

#### Interim summary

- ▶ In SCO, regularization is essential.
- ▶ In SCO, some learning rules might fail to generalize well.
- As a result, SCO provides a framework to differentiate between different algorithms.

We saw an instance where ERM fails if  $m \le O(d)$ . What about the output of algorithms? Can we prove similar lower bounds for the generalization ability of actual algorithms?



#### **Gradient Descent**

## Gradient Descent (full-batch)

initialize at 
$$w_1 \in \mathcal{W}$$
;

update 
$$w_{t+1} = \Pi_{\mathcal{W}}\left(w_t - \frac{\eta}{m}\sum_{i=1}^m \nabla_{w_t}f(w_t, z_i)\right), \ 1 \leq t < T;$$

return

$$\bar{w}_S := \frac{1}{S} \sum_{i=1}^S w_{T-i+1}.$$

where  $\Pi_{\mathcal{W}}$  denotes the projection onto the convex set  $\mathcal{W} \subset \mathbb{R}^d$ , and  $\eta > 0$  is the learning rate. The output is the average over the last *S* iterates.



#### **Stochastic Gradient Descent**

## Stochastic Gradient Descent

$$\begin{array}{ll} \text{initialize at} & w_1 \in \mathcal{W}; \\ \text{update} & w_{t+1} = \Pi_{\mathcal{W}} \Big( w_t - \eta \, \nabla_{w_t} f(w_t, z_i) \Big), \ 1 \leq t < T; \\ \text{return} & \bar{w}_S \coloneqq \frac{1}{S} \sum_{i=1}^S w_{T-i+1}. \end{array}$$



- Construction of a learning problem, where GD (starting from a data-independent initialization) outputs a bad ERM, unless trained with  $m = \Omega(\sqrt{d})$  samples.
- Construction of a learning problem, where SGD outputs an *underfit* model, unless trained with  $m = \tilde{\Omega}(\sqrt{d})$  samples.

Significance: Previous known lower bound was of the form  $m = \Omega(\log d)$  due to [AKL21].

#### Sample complexity of GD (Theorem)

### Theorem 3

Fix m > 0,  $T > 3200^2$  and  $0 \le \eta \le \frac{1}{5\sqrt{T}}$  and let  $d = 178mT + 2m^2 + \max\{1, 25\eta^2T^2\}$ . There exists a distribution  $\mathcal{D}$  over instance set  $\mathcal{Z}$  and a convex, differentiable and 1-Lipschitz loss function  $f : \mathbb{R}^d \times Z \to \mathbb{R}$  such that for GD (either projected or unprojected; with  $W = \mathbb{B}^d$  or  $W = \mathbb{R}^d$  respectively) initialized at  $w_1 = 0$  with step size  $\eta$ , for all t = 1, ..., T, the t-suffix averaged iterate has, with probability at least  $\frac{1}{6}$  over the choice of the training sample,

$$F(\bar{w}_{T,t}) - F(w^{\star}) = \Omega\left(\min\left\{\eta\sqrt{T} + \frac{1}{\eta T}, 1\right\}\right).$$

Introduction	Sample complexity of GD & SGD in SCO	Conclusion	Appendix
000	0000		
0000	0000		

#### Sample complexity of GD (Implication)

Recall:

$$F(w_{T,t}) - F(w^*) = \Omega\left(\min\left\{\eta\sqrt{T} + \frac{1}{\eta T}, 1\right\}\right).$$

• For 
$$T = m$$
 and  $\eta = \Theta(1/\sqrt{m})$ , we get:

$$F(w_{T,t})-F(w^{\star})=\Omega(1).$$

Furthermore,  $d = 178mT + 2m^2 + \max\{1, 25\eta^2 T^2\} = \Theta(m^2)$ which implies that at least  $m \ge \Omega(\sqrt{d})$  samples required for GD to reach nontrivial population loss.



▶ Replicate Feldman's construction in *T* orthogonal subspaces.

"Encode" the bad ERM into the weights.

Decode the bad ERM (from the gradients) and move towards it in each of the subspaces in a sequential order.

Introduction 000 0000	Sample complexity of GD & SGD in SCO ○○○○	Conclusion O	Appendix 00
0000			

Sample complexity of GD (Construction)

See board for an outline of the proof.

#### Sample of SGD (Theorem)

## A similar construction shows for SGD:

## Theorem 4

Fix m > 2048 and  $0 \le \eta \le \frac{1}{5\sqrt{m}}$  and let  $d = 712m \log m + 2m^2 + \max \{1, 25\eta^2 m^2\}$ . There exists a distribution  $\mathcal{D}$  over instance set  $\mathcal{Z}$  and a convex, 1-Lipschitz and differentiable loss function  $f : \mathbb{R}^d \times \mathcal{Z} \to \mathbb{R}$  such that for one-pass SGD (either projected or unprojected; with  $W = \mathbb{B}^d$  or  $W = \mathbb{R}^d$ respectively) over T = m steps initialized at  $w_1 = 0$  with step size  $\eta$ , for all  $t = 1, \ldots, T$ , the t-suffix averaged iterate has, with probability at least  $\frac{1}{2}$  over the choice of the training sample,

$$\hat{\mathcal{F}}(w_{\mathcal{T},t}) - \hat{\mathcal{F}}(\hat{w}_{\star}) = \Omega\left(\min\left\{\eta\sqrt{\mathcal{T}} + \frac{1}{\eta\mathcal{T}}, 1\right\}\right)$$

Introduction 000 0000	Sample complexity of GD & SGD in SCO	Conclusion ●	Appendix 00
Conclusion			

- ▶ In SCO, different rules than ERM need to be considered.
- [SSK24] show that GD requires at least  $\Omega(\sqrt{d})$  samples in order to generalize.
- In a follow-up work, [Liv24] improves this to Ω(d) samples, which is tight. Almost no benefit of GD over plain ERM in SCO!

When is GD provably better than a default ERM?

I Zaghloul Amir, Tomer Koren, and Roi Livni. Sgd generalizes better than gd (and regularization doesn't help). *ArXiv*, abs/2102.01117, 2021.

# Vitaly Feldman.

Generalization of erm in stochastic convex optimization: The dimension strikes back.

In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

# 📄 Roi Livni.

The sample complexity of gradient descent in stochastic convex optimization.

In The Thirty-eighth Annual Conference on Neural Information Processing Systems, 2024.

# Shai Shalev-Shwartz.

Online learning: Theory, algorithms, and applications. 08 2007.

- Matan Schliserman, Uri Sherman, and Tomer Koren. The dimension strikes back with gradients: Generalization of gradient methods in stochastic convex optimization. In 36th International Conference on Algorithmic Learning Theory, 2024.
- Shai Shalev-Shwartz, Ohad Shamir, Nathan Srebro, and Karthik Sridharan.

Learnability, stability and uniform convergence.

Journal of Machine Learning Research, 11(90):2635–2670, 2010.

Martin Zinkevich.

Online convex programming and generalized infinitesimal gradient ascent.

Introduction	
000	
0000	

In Proceedings of the Twentieth International Conference on International Conference on Machine Learning, ICML'03, page 928–935. AAAI Press, 2003.

Introduction 000 0000	Sample complexity of GD & SGD in SCO	Conclusion O	Appendix ●0

#### Feldman construction

- ▶ U: set of 2<sup>d</sup> unit vectors in  $\mathbb{R}^d$  such that  $|\langle u, v \rangle| \le 1/8$  for all  $u \ne v$ .
- ▶ Distribution  $\mathcal{D}$ : uniform over P(U).
- Sample *d* i.i.d. sets  $V_1, \ldots, V_d$ .
- ► Loss function: f<sub>F16</sub>(w, V) = max{1/2, max<sub>u∈V</sub>⟨w, u⟩} (convex and Lipschitz).
- ▶  $\mathbb{P}[\exists u_0 \in U : u_0 \notin V_i, \forall i \in [m]] = 1 (1 \frac{1}{2^d})^{2^d} \ge 1 e.$

Notice that  $u_0$  is an ERM and  $F_{F16}(u_0) - \hat{F}_{F16}(u_0) = \frac{1}{4} > 0$ . *Remark*: Existence of *U* via probabilistic method (dimension *d* large enough).

Introduction 000 0000	Sample complexity of GD & SGD in SCO	Conclusion O	Appendix ○●
GD lower l	bound construction		

- Hypothesis space:  $W = \mathbb{R}^d$ .
- ► Instance space: Z = P(U) × [m<sup>2</sup>], where U is the set of nearly orthogonal vectors.
- ▶ Distribution  $\mathcal{D}$ : uniform over  $P(U) \times [m^2]$ .
- ▶ Loss function  $f : W \times (P(U) \times [m^2]) \mapsto \mathbb{R}$  defined as:

$$f(w, (V, j)) = l_1(w, V) + l_2(w, (V, j)) + l_3(w) + l_4(w),$$

where:

▶ 
$$l_1(w, V) = \sqrt{\sum_{k=2}^{T} f_{F16'}^2(w^{(k)}, V)}, f_{F16'}(w, V) = \max\{\frac{3\eta}{32}, \max_{u \in V} \langle u, w \rangle\}.$$
  
▶  $l_2(w, (V, j)) = \langle -\phi(V, j), w^{(0)} \rangle, \phi : P(U) \times [m^2] \mapsto 2m^2.$   
▶  $l_3(w) = \max\{\delta_1, \max_{\psi \in \Psi}\{\langle \psi, w^{(0)} \rangle - \beta \langle \alpha(\psi), w^{(1)} \rangle\}\}.$   
▶  $l_4(w) = \max\{\delta_2, \max_{u \in U, k < T}\{\frac{3}{8}\langle u, w^{(k)} \rangle - \frac{1}{2}\langle u, w^{(k+1)} \rangle\}\}.$