Distributed Optimization for Machine Learning

Lecture 8 - Stochastic Gradient Methods

Tianyi Chen

School of Electrical and Computer Engineering
Cornell Tech, Cornell University

September 22, 2025

Revisit model training

Let's make our price predictor more realistic by adding more features.

Size (sq. ft.)	# Bedrooms	Age (years)	Price ($k)
1200	3	10	250
2000	4	5	350
800	2	25	150

Our goal is to find the best parameter $\mathbf{x}$ of model $h_{\mathbf{x}}(\mathbf{a})$ by minimizing

$$f(\mathbf{x}) = \frac{1}{2n} \sum_{i=1}^{n} (h_{\mathbf{x}}(\mathbf{a}^{(i)}) - y^{(i)})^2$$

Given previous discussion about $f(\mathbf{x})$, what is unique about this objective?

Empirical risk minimization - a "machine learning" name

Let $\{\mathbf{a}_i, y_i\}_{i=1}^n$ be n random samples, and consider

$$\min_{\mathbf{x}} \underbrace{F(\mathbf{x}) := \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{x}; \{\mathbf{a}_i, y_i\})}_{\text{empirical risk}}$$

e.g., quadratic loss $f(\mathbf{x}; \{\mathbf{a}_i, y_i\}) = (\mathbf{a}_i^\top \mathbf{x} - y_i)^2$. If one draws index

$j \sim \mathrm{Unif}(1, \ldots, n)$ uniformly at random, then

$$F(\mathbf{x}) = \mathbb{E}_j[f(\mathbf{x}; \{\mathbf{a}_j, y_j\})]$$

Stochastic programming - an "optimization" name

We view both the model training and testing problems as

$$\min_{\mathbf{x}} \quad \underbrace{F(\mathbf{x}) = \mathbb{E}[f(\mathbf{x};\xi)]}_{\text{expected risk, popular risk, ...}}$$

- ξ: randomness in problem
- suppose $f(\cdot;\xi)$ is convex for every ξ (and hence $F(\cdot)$ is convex)

Connecting the two views: Goal vs. Reality

We ideally want a model $\mathbf{x}$ that performs well on *all future data* $\mathcal{D}$:

$$\min_{\mathbf{x}} \; \mathbb{E}_{(\mathbf{a},y)\sim\mathcal{D}}[f(\mathbf{x};\{\mathbf{a},y\})]$$

We can't compute this because we don't have access to all data.

$$\Downarrow$$

We use our *finite training sample average* as a **proxy** for the true $\mathcal{D}$:

$$\frac{1}{n}\sum_{i=1}^{n} f(\mathbf{x};\{\mathbf{a}_i,y_i\}) \approx \mathbb{E}_{(\mathbf{a},y)\sim\mathcal{D}}[f(\mathbf{x};\{\mathbf{a},y\})]$$

A natural solution

Under "mild" technical conditions, if we run the gradient descent method from previous lectures, we have

$$\begin{aligned} \mathbf{x}^{t+1} &= \mathbf{x}^t - \eta_t \nabla F(\mathbf{x}^t) \\ &= \mathbf{x}^t - \eta_t \nabla \mathbb{E}[f(\mathbf{x}^t; \xi)] \\ &= \mathbf{x}^t - \eta_t \mathbb{E}[\nabla_{\mathbf{x}} f(\mathbf{x}^t; \xi)] \end{aligned}$$

Issues:

- testing setting - distribution of ξ may be unknown
- training setting - even if it is known, evaluating is expensive

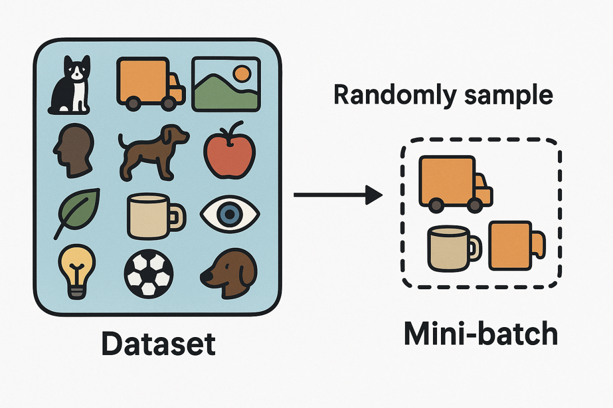
Why is expectation expensive?

- The expectation is over the entire data distribution:

$$\nabla F(\mathbf{x}) = \mathbb{E}_{\xi}[\nabla f(\mathbf{x}; \xi)]$$

- In practice, this means averaging gradients over **all samples**
- Example: ImageNet has > 1 million samples - one full gradient step would require computing 1,000,000+ gradients!
- **Takeaway:** Exact expectation is computationally infeasible; motivates stochastic approximations.

What should we do?



Generated by GPT 5 with prompt "generate one cartoon for samping"

Table of Contents

Stochastic gradient descent (SGD)

Stochastic gradient descent (SGD)

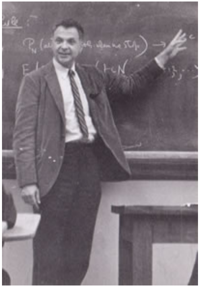
$$\mathbf{x}^{t+1} = \mathbf{x}^t - \eta_t g(\mathbf{x}^t; \xi^t) \tag{1}$$

where $g(\mathbf{x}^t; \xi^t)$ is an *unbiased estimate* of $\nabla F(\mathbf{x}^t)$, i.e.

$$\mathbb{E}[g(\mathbf{x}^t; \xi^t)] = \nabla F(\mathbf{x}^t)$$

— *Robbins, Monro '51*

Stochastic gradient descent (SGD)



Herbert Robbins

Stochastic approximation / Stochastic gradient descent (SGD)

$$\mathbf{x}^{t+1} = \mathbf{x}^{t} - \eta_t g(\mathbf{x}^{t}; \xi^{t}) \tag{1}$$

- a stochastic algorithm for finding a critical point $\mathbf{x}$ obeying $\nabla F(\mathbf{x}) = 0$
- more generally, a stochastic algorithm for finding the roots of $G(\mathbf{x}) := \mathbb{E}[g(\mathbf{x}; \xi)]$
- $\mathbf{x}$ does not necessarily obey $g(\mathbf{x}; \xi) = 0$

— *Robbins, Monro '51*

Example: SGD for empirical risk minimization

$$\min_{\mathbf{x}} \; F(\mathbf{x}) := \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{x}; \{\mathbf{a}_i, y_i\})$$

for $t = 0, 1, \ldots$

- choose i_t uniformly at random, and compute $\nabla_{\mathbf{x}} f_{i_t}(\mathbf{x}^t; \{\mathbf{a}_{i_t}, y_{i_t}\})$
- run the gradient descent update

$$\mathbf{x}^{t+1} = \mathbf{x}^t - \eta_t \nabla_{\mathbf{x}} f_{i_t}(\mathbf{x}^t; \{\mathbf{a}_{i_t}, y_{i_t}\})$$

end for

Compare GD and SGD trajectories

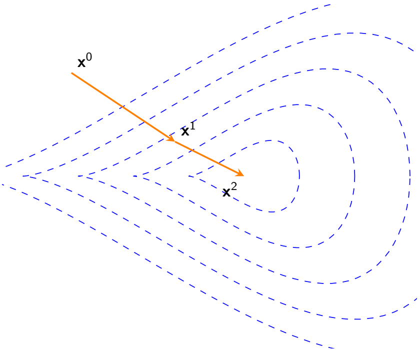


Figure: Example trajectory of Stochastic Gradient Descent (SGD) on a 2D loss landscape. The path is more erratic due to the noisy gradient estimates.

Example: SGD for empirical risk minimization

Benefits: SGD exploits information more efficiently than batch methods

- practical data usually involve lots of redundancy; using all data simultaneously in each iteration might be inefficient
- SGD is particularly efficient at the very beginning, as it achieves fast initial improvement with low per-iteration cost

Table of Contents

Strongly convex and smooth problems

$$\min_{\mathbf{x}} \ F(\mathbf{x}) := \mathbb{E}[f(\mathbf{x};\xi)]$$

Assumptions:

- F: μ-strongly convex, L-smooth
- $g(\mathbf{x}^t;\xi^t)$: an unbiased estimate of $\nabla F(\mathbf{x}^t)$ given $\{\xi^0,\ldots,\xi^{t-1}\}$
- $g(\mathbf{x}^t;\xi^t)$ has bounded variance: for all $\mathbf{x}$,

$$\mathbb{E}[||g(\mathbf{x};\xi)||_2^2] \le \sigma_g^2 + c_g||\nabla F(\mathbf{x})||_2^2 \tag{2}$$

Why having unbiasedness and bounded variance?

Mini-batch gradients: Variance reduction

Instead of using a single sample ξ^t, we can use a **mini-batch** of B i.i.d. samples $\{\xi_1^t, \dots, \xi_B^t\}$ to form a better gradient estimate:

$$g_B(\mathbf{x}^t) := \frac{1}{B}\sum_{i=1}^{B} g(\mathbf{x}^t; \xi_i^t)$$

- **Unbiasedness:** The mini-batch estimate is still unbiased.

$$\mathbb{E}[g_B(\mathbf{x}^t)] = \frac{1}{B}\sum_{i=1}^{B}\mathbb{E}[g(\mathbf{x}^t; \xi_i^t)] = \frac{1}{B}\sum_{i=1}^{B}\nabla F(\mathbf{x}^t) = \nabla F(\mathbf{x}^t)$$

- **Reduced variance:** The variance σ_g^2 is reduced by a factor of B:

$$\mathbb{E}[||g_B(\mathbf{x}; \xi)||_2^2] \leq \underbrace{\frac{\sigma_g^2}{B}}_{\text{Reduced!}} + \underbrace{\left(1 + \frac{c_g - 1}{B}\right)}_{c_B} ||\nabla F(\mathbf{x})||_2^2$$

Proof: Variance reduction

Let's expand the squared Euclidean norm of the mini-batch gradient:

$$\begin{aligned}
\|g_B(\mathbf{x})\|_2^2 &= \left\| \frac{1}{B} \sum_{i=1}^{B} g(\mathbf{x}; \xi_i) \right\|_2^2 \\
&= \frac{1}{B^2} \left\| \sum_{i=1}^{B} g(\mathbf{x}; \xi_i) \right\|_2^2 \\
&= \frac{1}{B^2} \left(\sum_{i=1}^{B} \|g(\mathbf{x}; \xi_i)\|_2^2 + \sum_{i \neq j} g(\mathbf{x}; \xi_i)^\top g(\mathbf{x}; \xi_j) \right)
\end{aligned}$$

Now, we take the expectation. By linearity of expectation:

$$\mathbb{E}[\|g_B(\mathbf{x})\|_2^2] = \frac{1}{B^2} \left(\sum_{i=1}^{B} \mathbb{E}[\|g(\mathbf{x}; \xi_i)\|_2^2] + \sum_{i \neq j} \mathbb{E}[g(\mathbf{x}; \xi_i)^\top g(\mathbf{x}; \xi_j)] \right)$$

Proof: Variance reduction

1. **Sum of squared norms:** Using i.i.d. and the individual variance:

$$\sum_{i=1}^{B} \mathbb{E}[\|g(\mathbf{x};\xi_i)\|_2^2] \leq B(\sigma_g^2 + c_g\|\nabla F(\mathbf{x})\|_2^2)$$

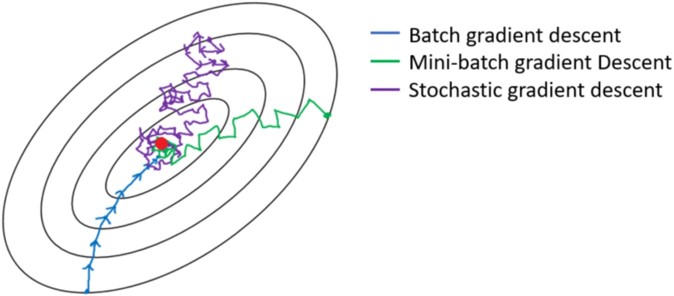
2. **Cross-terms:** For $i \neq j$, $\mathbb{E}[g(\mathbf{x};\xi_i)^\top g(\mathbf{x};\xi_j)] = \|\nabla F(\mathbf{x})\|_2^2$
   There are $B(B-1)$ such cross-terms.

$$\sum_{i \neq j} \mathbb{E}[g(\mathbf{x};\xi_i)^\top g(\mathbf{x};\xi_j)] = B(B-1)\|\nabla F(\mathbf{x})\|_2^2$$

Substitute these back into the expectation:

$$\begin{aligned}
\mathbb{E}[\|g_B(\mathbf{x})\|_2^2] &\leq \frac{1}{B^2}\left(B(\sigma_g^2 + c_g\|\nabla F(\mathbf{x})\|_2^2) + B(B-1)\|\nabla F(\mathbf{x})\|_2^2\right) \\
&= \frac{B\sigma_g^2}{B^2} + \frac{Bc_g\|\nabla F(\mathbf{x})\|_2^2}{B^2} + \frac{(B^2 - B)\|\nabla F(\mathbf{x})\|_2^2}{B^2} \\
&= \frac{\sigma_g^2}{B} + \left(1 + \frac{c_g - 1}{B}\right)\|\nabla F(\mathbf{x})\|_2^2
\end{aligned}$$

Compare GD, SGD and mini-batch SGD trajectories



Convergence: fixed stepsizes

Theorem 1 (Strong convexity and fixed stepsizes)

Under the assumptions in previous slide, if $\eta_t \equiv \eta \leq \frac{1}{Lc_g}$, then SGD (1) achieves

$$\mathbb{E}[F(\mathbf{x}^t) - F(\mathbf{x}^*)] \leq \frac{\eta L \sigma_g^2}{2\mu} + (1 - \eta\mu)^t (F(\mathbf{x}^0) - F(\mathbf{x}^*))$$

- check Bottou, Curtis, Nocedal '18 (Theorem 4.6) for the proof

"Optimization methods for large-scale machine learning," Bottou, Curtis, Noceda, arXiv, 2018.

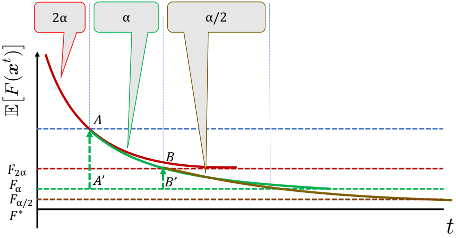
Implications: SGD with fixed stepsizes

$$\mathbb{E}[F(\mathbf{x}^t) - F(\mathbf{x}^*)] \le \frac{\eta L \sigma_g^2}{2\mu} + (1 - \eta\mu)^t (F(\mathbf{x}^0) - F(\mathbf{x}^*))$$

- fast (linear) convergence at the very beginning
- converges to some neighborhood of $\mathbf{x}^*$ - variation in gradient computation prevents further progress
- when gradient computation is noiseless (i.e. $\sigma_g = 0$), it converges linearly to optimal points
- smaller stepsizes η yield better converging points

One practical strategy

Run SGD with fixed stepsizes; whenever progress stalls, reduce stepsizes and continue SGD.



Bottou, Curtis, Nocedal '18

whenever progress stalls, we half the stepsizes and repeat

Convergence with diminishing stepsizes

Theorem 2 (Strong convexity and diminishing stepsizes)

Suppose F is μ-strongly convex, and (2) holds with $c_g = 0$. If $\eta_t = \frac{\theta}{t+1}$ for some $\theta > \frac{1}{2\mu}$, then SGD (1) achieves

$$\mathbb{E}[||\mathbf{x}^t - \mathbf{x}^*||_2^2] \leq \frac{c_\theta}{t+1}$$

where $c_\theta = \max\left\{\frac{2\theta^2\sigma_g^2}{2\mu\theta - 1}, ||\mathbf{x}_0 - \mathbf{x}^*||_2^2\right\}$.

- convergence rate $\mathcal{O}(1/t)$ with diminishing stepsize $\eta_t \approx 1/t$

Proof of Theorem 2

Using the SGD update rule, we have (compare with GD proof steps)

$$\begin{aligned}||\mathbf{x}^{t+1} - \mathbf{x}^*||_2^2 &= ||\mathbf{x}^t - \eta_t g(\mathbf{x}^t; \xi^t) - \mathbf{x}^*||_2^2 \\ &= ||\mathbf{x}^t - \mathbf{x}^*||_2^2 - 2\eta_t (\mathbf{x}^t - \mathbf{x}^*)^\top g(\mathbf{x}^t; \xi^t) + \eta_t^2 ||g(\mathbf{x}^t; \xi^t)||_2^2 \ (\star)\end{aligned}$$

Since $\mathbf{x}^t$ is independent of ξ_t, apply the law of total expectation to obtain

$$\begin{aligned}\mathbb{E}[(\mathbf{x}^t - \mathbf{x}^*)^\top g(\mathbf{x}^t; \xi^t)] &= \mathbb{E}[\mathbb{E}[(\mathbf{x}^t - \mathbf{x}^*)^\top g(\mathbf{x}^t; \xi^t) | \xi_1, \ldots, \xi_{t-1}]] \\ &= \mathbb{E}[(\mathbf{x}^t - \mathbf{x}^*)^\top \mathbb{E}[g(\mathbf{x}^t; \xi^t) | \xi_1, \ldots, \xi_{t-1}]] \\ &= \mathbb{E}[(\mathbf{x}^t - \mathbf{x}^*)^\top \nabla F(\mathbf{x}^t)] \qquad (\diamond)\end{aligned}$$

Proof of Theorem 2 (cont.)

Furthermore, strong convexity gives

$$\langle \nabla F(\mathbf{x}^t), \mathbf{x}^t - \mathbf{x}^* \rangle = \langle \nabla F(\mathbf{x}^t) - \underbrace{\nabla F(\mathbf{x}^*)}_{\mathbf{0}}, \mathbf{x}^t - \mathbf{x}^* \rangle \geq \mu \|\mathbf{x}^t - \mathbf{x}^*\|_2^2$$

$$\implies \mathbb{E}[\langle \nabla F(\mathbf{x}^t), \mathbf{x}^t - \mathbf{x}^* \rangle] \geq \mu \mathbb{E}[\|\mathbf{x}^t - \mathbf{x}^*\|_2^2]$$

Combine the above inequalities and (2) (with $c_g = 0$) to obtain

$$\mathbb{E}[\|\mathbf{x}^{t+1} - \mathbf{x}^*\|_2^2] \leq (1 - 2\mu\eta_t)\mathbb{E}[\|\mathbf{x}^t - \mathbf{x}^*\|_2^2] + \underbrace{\eta_t^2 \sigma_g^2}_{\text{does not vanish unless } \eta_t \to 0}$$

Take $\eta_t = \frac{\theta}{t+1}$ and use induction to conclude the proof (exercise!)

Optimality*

— *Nemirovski, Yudin '83, Agarwal et al. '11, Raginsky, Rakhlin '11*

- Informally, when minimizing strongly convex functions, no algorithm performing t queries to noisy first-order oracles can achieve an accuracy better than the order of $1/t$.

$\Rightarrow$ SGD with stepsizes $\eta_t \approx 1/t$ is optimal.

Optimality*

— Nemirovski, Yudin '83

More precisely, consider a class of problems in which f is μ-strongly convex and L-smooth, and $Var(\|g(\mathbf{x}^t;\xi^t)\|_2) \le \sigma^2$. Then the worst-case iteration complexity for (stochastic) first-order methods:

$$\sqrt{\frac{L}{\mu}}\log\left(\frac{L\|\mathbf{x}_0-\mathbf{x}^*\|_2^2}{\epsilon}\right)+\frac{\sigma^2}{\mu\epsilon}$$

- *for deterministic cases: $\sigma = 0$, and hence the lower bound is*

$$\sqrt{\frac{L}{\mu}}\log\left(\frac{L\|\mathbf{x}_0-\mathbf{x}^*\|_2^2}{\epsilon}\right)$$

(achievable by Nesterov's method)

Optimality*

— *Nemirovski, Yudin '83*

More precisely, consider a class of problems in which f is μ-strongly convex and L-smooth, and Var$(||g(\mathbf{x}^t; \xi^t)||_2) \leq \sigma^2$. Then the worst-case iteration complexity for (stochastic) first-order methods:

$$\sqrt{\frac{L}{\mu}} \log\left(\frac{L||\mathbf{x}_0 - \mathbf{x}^*||_2^2}{\epsilon}\right) + \frac{\sigma^2}{\mu\epsilon}$$

- *for noisy cases with large σ, the lower bound is dominated by*

$$\frac{\sigma^2}{\mu} \cdot \frac{1}{\epsilon}$$

Comparisons with batch GD

Empirical risk minimization with n samples:

	iteration complexity	**per-iteration cost**	**total comput. cost**
batch GD	$\log \frac{1}{\epsilon}$	n	$n \log \frac{1}{\epsilon}$
SGD	$\frac{1}{\epsilon}$	1	$\frac{1}{\epsilon}$

SGD is more appealing for large n and moderate accuracy ϵ (in which case $\frac{1}{\epsilon} < n \log \frac{1}{\epsilon}$)

$\Rightarrow$ which often arises in the *big data* regime!

Convex problems

What if we lose strong convexity?

$$\min_{\mathbf{x}} \; F(\mathbf{x}) := \mathbb{E}[f(\mathbf{x};\xi)]$$

Assumptions:

- F: convex
- $\mathbb{E}[||g(\mathbf{x};\xi)||_2^2] \leq \sigma_g^2$ for all $\mathbf{x}$
- $g(\mathbf{x}^t;\xi^t)$ is an unbiased estimate of $\nabla F(\mathbf{x}^t)$ given $\{\xi^0, \ldots, \xi^{t-1}\}$

Convex problems

Suppose we return a weighted average

$$\tilde{\mathbf{x}}^t := \sum_{k=0}^{t} \frac{\eta_k}{\sum_{j=0}^{t} \eta_j} \mathbf{x}^k$$

Theorem 3 Under the assumptions in the previous slide, then

$$\mathbb{E}[F(\tilde{\mathbf{x}}^t) - F(\mathbf{x}^*)] \le \frac{\frac{1}{2}\mathbb{E}[||\mathbf{x}^0 - \mathbf{x}^*||_2^2] + \frac{1}{2}\sigma_g^2 \sum_{k=0}^{t} \eta_k^2}{\sum_{k=0}^{t} \eta_k}$$

- if $\eta_t \approx 1/\sqrt{t}$, then

$$\mathbb{E}[F(\tilde{\mathbf{x}}^t) - F(\mathbf{x}^*)] \le \frac{\log t}{\sqrt{t}}$$

Proof of Theorem 3

By convexity of $F(\mathbf{x})$, we have $F(\mathbf{x}) \geq F(\mathbf{x}^t) + (\mathbf{x} - \mathbf{x}^t)^\top \nabla F(\mathbf{x}^t)$

$$\implies \mathbb{E}[(\mathbf{x}^t - \mathbf{x}^*)^\top \nabla F(\mathbf{x}^t)] \geq \mathbb{E}[F(\mathbf{x}^t) - F(\mathbf{x}^*)]$$

This together with $(\star)$ and $(\diamond)$ in Proof of Theorem 2 implies

$$2\eta_k \mathbb{E}[F(\mathbf{x}^k) - F(\mathbf{x}^*)] \leq \mathbb{E}[||\mathbf{x}^k - \mathbf{x}^*||_2^2] - \mathbb{E}[||\mathbf{x}^{k+1} - \mathbf{x}^*||_2^2] + \eta_k^2 \sigma_g^2$$

Sum over $k = 0, \ldots, t$ to obtain

$$\sum_{k=0}^{t} 2\eta_k \mathbb{E}[F(\mathbf{x}^k) - F(\mathbf{x}^*)] \leq \mathbb{E}[||\mathbf{x}^0 - \mathbf{x}^*||_2^2] - \mathbb{E}[||\mathbf{x}^{t+1} - \mathbf{x}^*||_2^2] + \sigma_g^2 \sum_{k=0}^{t} \eta_k^2$$

$$\leq \mathbb{E}[||\mathbf{x}^0 - \mathbf{x}^*||_2^2] + \sigma_g^2 \sum_{k=0}^{t} \eta_k^2$$

Proof of Theorem 3 (cont.)

Setting $v_t = \frac{\eta_t}{\sum_{k=0}^{t} \eta_k}$ yields

$$\sum_{k=0}^{t} v_k \mathbb{E}[F(\mathbf{x}^k) - F(\mathbf{x}^*)] \le \frac{\frac{1}{2}\mathbb{E}[||\mathbf{x}^0 - \mathbf{x}^*||_2^2] + \frac{1}{2}\sigma_g^2 \sum_{k=0}^{t} \eta_k^2}{\sum_{k=0}^{t} \eta_k}$$

By convexity of $F(\mathbf{x})$, we arrive at

$$\begin{aligned}
\mathbb{E}[F(\tilde{\mathbf{x}}^t) - F(\mathbf{x}^*)] &\le \sum_{k=0}^{t} v_k \mathbb{E}[F(\mathbf{x}^k) - F(\mathbf{x}^*)] \\
&\le \frac{\frac{1}{2}\mathbb{E}[||\mathbf{x}^0 - \mathbf{x}^*||_2^2] + \frac{1}{2}\sigma_g^2 \sum_{k=0}^{t} \eta_k^2}{\sum_{k=0}^{t} \eta_k}
\end{aligned}$$

Recap and fine-tuning

- What we have talked about today?
  - $\Rightarrow$ Why we need SGD and how it works?
  - $\Rightarrow$ What is its convergence properties?

Welcome anonymous survey!

Reference

- "A stochastic approximation method," H. Robbins, S. Monro, The Annals of Mathematical Statistics, 1951.
- "Robust stochastic approximation approach to stochastic programming," A. Nemirovski et al., SIAM Journal on Optimization, 2009.
- "Optimization methods for large-scale machine learning," L. Bottou et al., arXiv, 2016.
- "New stochastic approximation type procedures," B. Polyak, Automat. Remote Control, 1990.
- "Acceleration of stochastic approximation by averaging," B. Polyak, A. Juditsky, SIAM Journal on Control and Optimization, 1992.
- "First-order methods in optimization," A. Beck, Vol. 25, SIAM, 2017.
- "A convergence theorem for nonnegative almost supermartingales and some applications," H. Robbins, D. Siegmund, Optimizing methods in statistics, 1971.