Optimization Algorithms for Training Over-Parameterized Models

Mark Schmidt

University of British Columbia

Joint work with: Francis Bach, Gauthier Gidel, Simon LaCoste-Julien, Jonathan Lavington, Frederik Kunstner, Issam Laradji, Aaron Mishkin, Si Yi Meng, Nicolas Le Roux, and Sharan Vaswani

Motivation: Over-Parameterized Models in Machine Learning

- Modern machine learning practioners often do a weird thing:
    - Train (and get excellent performance) with models that are over-parameterized.
        - "The model is so complicated that you can fit the data perfectly".
        - The exact setting where we normally teach students that bad overfitting happens.

- Examples:
    - Many state-of-the-art deep computer vision models are over-parameterized.
        - Models powerful enough to fit training set with random labels [Zhang et al., 2017].
    - Linear models with sufficiently expressive features [Liang & Rakhlin, 2018].

- Many recent papers study benefits of over-parameterization in various settings:
    - Algorithms may have implicit regularization that reduces overfitting.
    - Optimizers may find global optima in problems we normally view as hard.

Single-Slide Summary of this Talk

- For over-parameterized models, you need to re-think how optimization works!

1. Stochastic gradient descent converges faster for over-parameterized models.
    - May help explain the empirical success of constant step sizes in practice.
    - May help explain why it has been so difficult to develop faster algorithms.
2. We can design faster stochastic algorithms for over-parameterized models.
    - Over-parameterization allows Nesterov acceleration and second-order methods.
    - Over-parameterization allows better sampling schemes and tighter regret bounds.
3. We can design stochastic algorithms that are easier to use:
    - Algorithms that do not depend on problem-dependent constants.
    - Algorithms that adapt to the difficulty of the problem.

Outline

Quick SGD Overview

- The stochastic gradient descent (SGD) method uses iterations of the form

$$w_{k+1} = w_k - \alpha_k \nabla f(w_k, z_k),$$

where α_k is the step size and z_k noise in the gradient.

  - Here we are trying to minimize a differentiable funciton f with parameters w.

- Classic analyses of SGD assume that the gradient approximation is unbiased,

$$\mathbb{E}[\nabla f(w_k, z_k)] = \nabla f(w^k),$$

and bound variation in noise in some way, like assuming for some σ^2 that

$$\mathbb{E}[\|\nabla f(w_k, z_k) - \nabla f(w^k)\|^2] \leq \sigma^2.$$

Special Case of Finite Sums

- SGD is often used to minimize functions f having a finite-sum structure,

$$f(w) = \frac{1}{n}\sum_{i=1}^{n} f_i(w),$$

where each f_i measures the error on training example i.

- SGD iterations chooses a random i_k to give an unbiased gradient approximation,

$$w_{k+1} = w_k - \alpha_k \nabla f_{i_k}(w^k),$$

- Key advantage for finite-sum problems: iteration cost is $O(1)$ in terms of n.
  - Warning: results will be stated for finite sums, but most apply to general noise.

Assumptions on the Function

- This talk will assume that ∇f is L-Lipschitz continuous (L-smooth),

$$\|\nabla f(w) - \nabla f(v)\| \leq L\|w - v\|,$$

and that each each ∇f_i is L_i-Lipschitz continuous (L_i-smooth).

$$\|\nabla f_i(w) - \nabla f_i(v)\| \leq L_i\|w - v\|.$$

- We use $L_{\max}$ as the maximum value of L_i across all examples.

- We will consider various restrictions on the growth of the function:

$$f(w) \geq f(v) + \langle \nabla f(v), w - v\rangle + \frac{\mu}{2}\|w - v\|^2 \qquad \text{(Strongly Convex)}$$

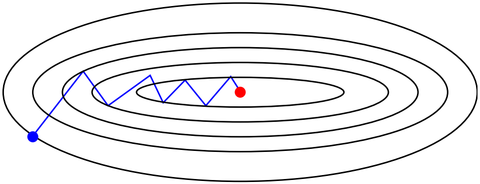
$$f(w) \geq f(v) + \langle \nabla f(v), w - v\rangle \qquad \text{(Convex)}$$

$$f(w) \geq f^* \qquad \text{(Bounded Below)}$$

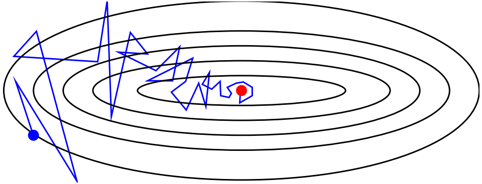
$$\frac{\mu}{2}\|\nabla f(w)\|^2 \geq f(w) - f^* \qquad \text{(PL inequality)}$$

Determnistic vs. Stochastic Gradient Descent

- Deterministic gradient descent converges with a small-enough constant step size.
  - Under any of the growth conditions.



- SGD needs a decreasing sequence of step sizes α_k to converge.
  - Under any of the growth conditions, for unbiased+(bounded variance) noise.

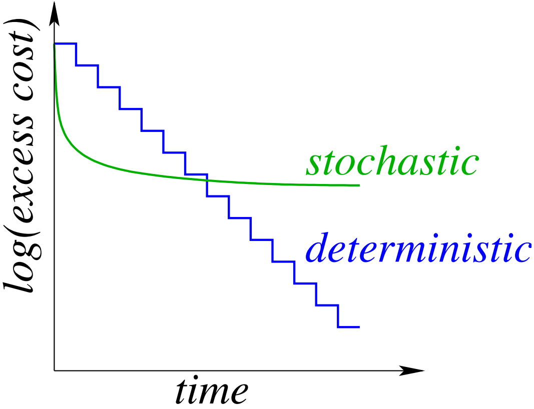


Classic Stochastic and Deterministic Convergence Rates

- After k iterations, SGD finds a w satisfying the following convergence rates:
  - $\mathbb{E}[f(w)] - f^* = O(1/k)$ for strongly-convex and PL functions.
  - $\mathbb{E}[f(w)] - f^* = O(1/\sqrt{k})$ for convex functions.
  - $\mathbb{E}[\|\nabla f(w)\|^2] = O(1/\sqrt{k})$ for bounded-below functions (which may be non-convex).
- These rates are slower than for deterministic gradient descent (where $\sigma^2 = 0$):
  - $f(w) - f^* = O(\gamma^k)$ for strongly-convex and PL functions (for some $\gamma < 1$).
  - $f(w) - f^* = O(1/k)$ for convex functions.
  - $\|\nabla f(w)\|^2 = O(1/k)$ for bounded-below functions.
- All deterministic results be achieved with small-enough constant step size α_k.
  - Deterministic method adapt to problem: do not need to know if f is convex/PL.

Classic Stochastic and Deterministic Convergence Rates

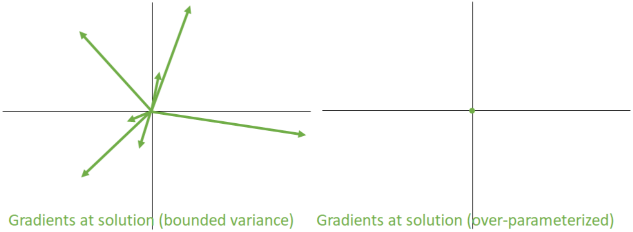
- Deterministic vs. stochastic gradient descent for strongly-convex/PL functions:



- Deterministic has linear convergence $O(\gamma^k)$ but $O(n)$ iteration cost.
- Stochastic has sublinear convergence $O(1/k)$ but $O(1)$ iteration cost.

Effect of Over-Parameterization on SGD

- We say a model is over-parameterized if it can exactly fit all training examples.
  - Unlike usual bounded variance assumption, we have $\nabla f_i(w_*) = 0$ for all i:



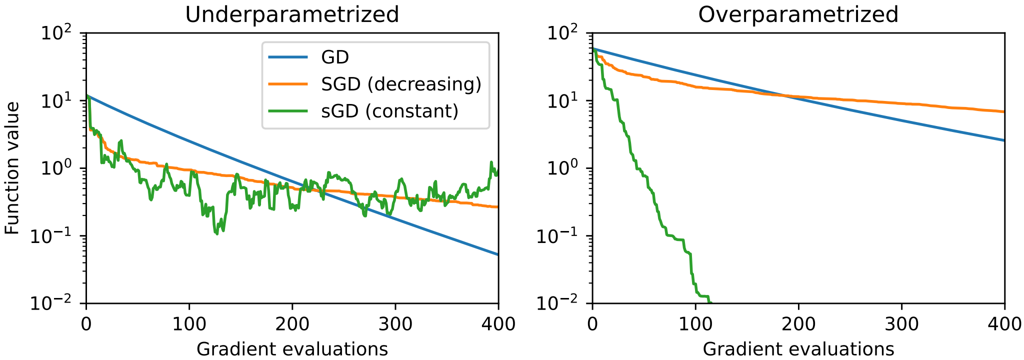
- Under over-parameterized models, the variance is 0 at minimizers.
  - And SGD converges with a sufficiently small constant step size.

Stochastic Convergence Rates under Over-Parameterization

- Over-parameterized: SGD can achieve the deterministic convergence rates,
  - $\mathbb{E}[f(w)] - f^* = O(\gamma^k)$ for strongly-convex and PL functions (for some $\gamma < 1$).
  - $\mathbb{E}[f(w)] - f^* = O(1/k)$ for convex functions.
  - $\mathbb{E}[\|\nabla f(w)\|^2] = O(1/k)$ for bounded-below functions (which may be non-convex).
- All of these above rates are obtained for any sufficiently small step size.
  - So SGD adapts to the difficulty of the problem.
    - The same step size works for strongly-convex and non-convex problems.
  - Partial explanation for the success of constant step sizes in practice.
    - Which do not converge in the usual setting.

Stochastic Convergence Rates under Over-Parameterization

- Comparison of least squares performance in under-/over-parameterized models:



Ways to Characterize Over-Parameterization

- First over-parameterization results are due to Solodov [1998] and Tseng [1998].
  - They considered variation on what is now called the strong growth condition (SGC),

$$\mathbb{E}[\|\nabla f_i(w)\|^2] \leq \rho \|\nabla f(w)\|^2.$$

- Bach & Moulines [2011] later analyze SGD when variance at solution is 0.
  - We call this the interpolation property (which is implied by the SGC),

$$\mathbb{E}[\|\nabla f_i(w_*)\|^2] = 0.$$

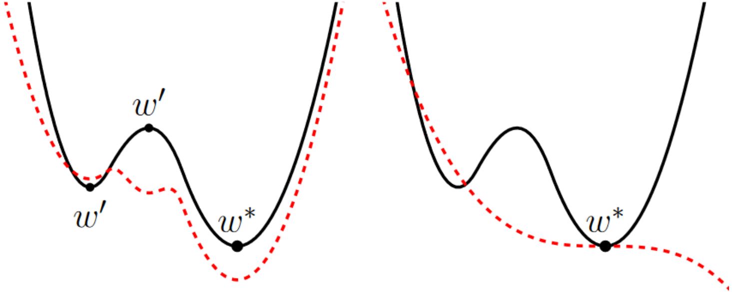
- An alternate condition was considerd by Vaswani et al. [2019].
  - The weak growth condition (WGC) for an L-smooth function is

$$\mathbb{E}[\|\nabla f_i(w)\|^2] \leq 2\rho L(f(w) - f(w_*)).$$

  - Relation between conditions for L-smooth f and L_{max}-smooth f_i:
    - SGC $\rightarrow$ interpolation and WGC.
    - For invex functions: interpolation $\rightarrow$ WGC.
    - For PL functions: WGC $\rightarrow$ SGC.

Strong Growth Condition vs. Weak Growth Condition

- SGC implies each f_i is stationary when f is stationary.
- Interpolation and WGC imply each f_i is stationary at global minizers.



  - Neither condition rules out non-isolated or multiple global minimizers.
- The constant under WGC may be smaller:
  - For PL functions satisfying SGC we have $\rho \leq L_{\text{max}}/\mu$.
  - For invex functions satisfying WGC we have $\rho \leq L_{\text{max}}/L$.

Over-Parameterized Results for Basic SGD with Constant Step

- Timeline of results for SGD with constant step size:

Solodov/Tseng [1998]	SGC	Asymptotic (rate on epochs)
Bach & Moulines [2011]	Interpolation	Strongly-convex (slow rate)
S. & Le Roux [2013]	SGC	Strongly-convex
S. & Le Roux [2013]	SGC	Convex
Needell et al. [2014]	Interpolation	Strongly-convex
Bassily et al. [2018]	Interpolation	PL (slow rate)
Vaswani et al. [2019]	SGC	PL
Vaswani et al. [2019]	SGC	Bounded Below
Vaswani et al. [2019]	WGC	Strongly-convex
Vaswani et al. [2019]	WGC	Convex

Example: Function Decrease under the SGC

- If we write the SGD step as a deterministic gradient descent step with error,

$$w^{k+1} = w^k - \alpha_k(\nabla f(w^k) + e^k),$$

  then under the SGC we can bound the expected error compared to the gradient,

$$\mathbb{E}[\|e^k\|^2] \leq (\rho - 1)\|\nabla f(w^k)\|^2,$$

- Recall the descent lemma for L-smooth f,

$$f(w^{k+1}) \leq f(w^k) + \langle \nabla f(w^k), w^{k+1} - w^k \rangle + \frac{L}{2}\|w^{k+1} - w^k\|^2.$$

- Under the SGC and a step size of $\alpha_k = 1/L\rho$ we obtain after simplifying that

$$\mathbb{E}[f(w^{k+1})] \leq f(w^k) - \frac{1}{2L\rho}\|\nabla f(w^k)\|^2,$$

  the function decrease of deterministic gradient descent up to a factor of ρ.
    - From this inequality you can derive the rates under the different assumptions.

Over-Parameterization vs. Advanced SGD Methods

- Variance-reduced SGD also speeds up the convergence of SGD for finite sums.
  - Though these rates depend on number of training examples n.

- For strongly-convex functions:
  - SAG[A] and SVRG require $\tilde{O}\left(\frac{L_{\text{max}}}{\mu} + n\right)$ iterations to reach accuracy ϵ.
  - SGD under WGC requires $\tilde{O}\left(\frac{L_{\text{max}}}{\mu}\right)$ iterations to reach accuracy ϵ.
  - Helps explain lack of improvement from variance-reduced methods on deep networks.

    [Defazio & Bottou, 2019]

- Specialized non-convex stochastic methods improve classic SGD rate.

  [Allen-Zhu, 2017].
  - But SGD non-convex rate under SGC has a better dependence on ϵ.

But my models are not over-parameterized!

- Various “close to over-parameterized” conditions exist.
  - Cevher & Vu [2017] analyze a generalization of the SGC

$$\mathbb{E}[\|\nabla f_i(w)\|^2] \leq \rho\|\nabla f(w)\|^2 + \sigma^2,$$

    which appears in earlier works like Polyak & Tsypkin [1973].
  - Bach & Moulines' [2011] result analyze a generalization of interpolation,

$$\mathbb{E}[\|\nabla f_i(w_*)\|^2] \leq \sigma^2,$$

    which is related to conditions in earliers works like Polyak & Juditsky [1992].
  - Gower et al. [2019] analyze expected smoothness which generalizes the WGC.

- These conditions are not sufficient for convergence with a constant step size.
  - But many of the ideas in this talk may still be useful.
  - Constant step size α still converges quickly to region of size $O(\alpha\sigma^2)$.
    - If σ^2 is small, this may be all you need.
    - And note that σ^2 decreases with the batch size.

Outline

Accelerated SGD for Over-Parameterized Models?

- Over-parameterization leads to faster convergence rates for SGD.
- But can we exploit over-parameterization to develop faster methods than SGD?
- For example, could we develop an accelerated SGD method?
  - Known that Nesterov acceleration improves empirical performance in some settings.

    [Sutskever et al., 2013]
- What about better sampling, second-order methods, regret bounds, and so on?

Review of Deterministic vs. Stochastic Acceleration

- For deterministic gradient descent:
  - Acceleration improves iteration complexity from $\tilde{O}(\kappa)$ to $\tilde{O}(\sqrt{\kappa})$.
    - Where $\kappa = L/\mu$.
- For stochastic gradient with bounded variance σ^2:
  - Acceleration could improve from $\tilde{O}\left(\frac{\sigma^2}{\mu\epsilon} + \kappa\right)$ to $\tilde{O}\left(\frac{\sigma^2}{\mu\epsilon} + \sqrt{\kappa}\right)$.
    - This is only faster if $\kappa > \sigma^2/\mu\epsilon$.
    - Otherwise, the variance term dominates and no acceleration.
- For variance-reduced stochastic gradient for finite-sum problems:
  - Acceleration improves from $\tilde{O}(n + \kappa)$ to $\tilde{O}(n + \sqrt{n\kappa})$.
    - This is only faster if $\kappa > n$.
    - Otherwise, the number of examples n dominates and no acceleration.

Stochastic Acceleration under Over-Parameterization

- In Vaswani et al. [2019], we presented an accelerated SGD under the SGC:

$$
\begin{aligned}
w_{k+1} &= y_k - \alpha_k \nabla f(y_k, z_k) \\
y_k &= \theta_k v_k + (1-\theta_k) w_k \\
v_{k+1} &= \beta_k v_k + (1-\beta_k) y_k - \gamma_k \nabla f(y_k, z_k).
\end{aligned}
$$

- For appropriate choices of $\{\alpha_k, \beta_k, \gamma_k, \theta_k\}$:
  - Acceleration improves complexity from $\tilde{O}(\rho\kappa)$ to $\tilde{O}(\rho\sqrt{\kappa})$.
    - Paper also includes an accelerated $O(\rho\sqrt{L/\epsilon})$ rate for convex functions.
- Related work:
  - Jain et al. [2018] had earlier given an accelerated method for least squares.
  - Liu & Belkin [2020] give accelerated method under interpolation beyond quadratics.
    - Show that accelerated SGC rates may be slower than non-accelerated WGC rates.
  - Mishkin [2020] improves the rate under SGC to $O(\sqrt{\rho\kappa})$ and $O(\sqrt{\rho L/\epsilon})$.
    - Faster than non-accelerated rates under interpolation/WGC.

Faster Sampling Strategies under Over-Parameterization

- Another way to speed up SGD is by changing the sampling strategy.

- Needell et al. [2014] consider non-uniform sampling:
  - Bias sampling distribution towards Lipschitz constants of individual examples.
  - Leads to a rate depending on average Lipschitz constant instead of maximum.
    - In classic SGD setting, only improves rate under certain conditions.

- Needel & Ward [2016] and Ma et al. [2018] consider mini-batch sampling.
  - Show that improves rate if we have parallel computation.
    - Support for “linear scaling rule” used in neural networks, and shows its limit.
  - Gower et al. [2019] analyze general sampling strategies under over-parameterization.

- HaoChen and Sra [2019] consider random shuffling of training examples.
  - Show that random shuffling converge at least as fast as uniform sampling.

Second-Order Stochastic Methods

- Can we speed up SGD using second-order updates?
  - In classic SGD setting, second-order updates do not improve $O(1/k)$ rate.
- Classic result for deterministic Gauss-Newton:
  - Achieves superlinear convergence under interpolation.
- Gürbüzbalaban et al. [2014] consider second-order methods with cyclic selection:
  - Show linear rate under SGC for Newton and Gauss-Newton.
- Meng et al. [2020] consider stochastic selection under the SGC:
  - Show superlinear rate with exponentially-growing batch size.
    - Previous works required faster-than-exponential growing batch size.
  - Includes self-concordant analysis, L-BFGS analysis, and Hessian-free implementation.

Other Over-Parameterization Results

- Cevher & Vu [2017] consider constrained optimization.
  - Show fast rates for projected stochastic gradient under a generalization of SGC.
- Fang et al. [2021] consider non-smooth optimization
  - Show fast rates for stochastic subgradient under a generalization of interpolation.
- Online learning methods are often analyzed in terms of regret.
  - For online convex optimization, SGD achieves regret of $O(\sqrt{k})$.
    - Using a decreasing sequence of step sizes.
  - Under interpolation, Orabona [2019] shows this can be reduced to $O(1)$.
    - Constant regret with a constant step size.
- Several recent works have considerd online imitation learning.
  - Yan et al. [2021] show that over-parameterization gives faster rate.
  - Lavington et al. [2022] show constant regret in very-general setting.

Outline

Setting the Step Size

- Unfortunately, these faster rates have a serious practical issue.
  - They are sensitive to the choice of step size (which depend on L and/or μ).
  - Performance significantly degrades under a poor choice of step size.

- You could search over several plausible guesses for the step size.
  - But searching is slow and a fixed step size may be sub-optimal anyways.
  - It would be better to adapt the step size as you go.

- In Vaswani et al. [2019], we consider a simple stochastic line search.
  - Achieves fast convergence rates in a variety of over-parameterized settings.
  - Outperforms a variety of methods in practice on many standard benchmarks.
    - In practice, cost is less than trying out 2 guesses for the step size.

Related Work - Without Over-Parameterization

- These exists a huge literature on setting the SGD step size.
  - Methods that adjust the step size as we go.
    - Keston [1958], Delyon & Juditsky [1993], Kushner & Yang [1995], Schaul et al. [2013], Schoenauaer-Sebag [2017], Rolinek & Martiu [2018].
  - Methods that "do gradient descent on the step size".
    - Sutton [1992], Almeida [1998], Schraudolph [1999], Shao & Yip [2000], Plagianakos et al. [2001], Gunes Baydin et al. [2018].
  - "Adaptive" methods like AdaGrad and its variations
    - Duchi et al. [2011], Zeiler [2012], King & Ba [2015], Luo et al. [2019], Reddi et al. [2019].
  - Coin betting methods.
    - Orabona & Tommasi [2017].
- None of these methods achieve faster rates possible in over-parameterized setting.

Related - Stochastic Line Searches

- A variety of works propose stochastic line-search or trust-region methods.
  - Friedlander and S. [2012], Byrd et al. [2012], Krejić and Krklec [2013], De et al. [2016], Gratton et al. [2017]. Mahsereci & Hennig [2017], Paquette & Scheinberg [2018], Blanchet et al. [2019].
- Without over-parameterization, require growing batch size for convergence.
- Tseng [98] proposes a stochastic line search in the over-parameterized setting.
  - Motivated by training neural networks.
  - Showed linear rate on epochs (complicated conditions).
  - Never widely-adopted and more complicated than our simple line search.
- Recent works evaluated Armijo-style stochastic line-search for deep learning.
  - Strong empirical performance for benchmark problems.
  - Theory requires growing batch size or only considers deterministic method.

[Bollapragada et al, 2018, Truong & Nguyen, 2018]

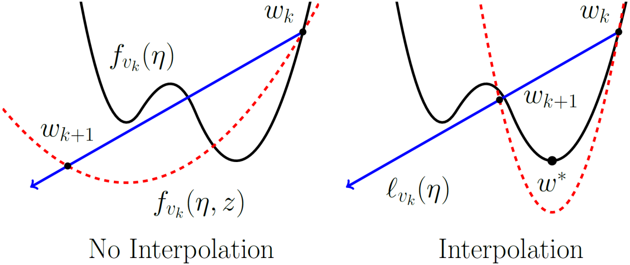
Stochastic Line Search - Theory

- An Armijo line-search on the mini-batch selects a step size satisfying

$$f_{i_k}(w_k - \alpha_k \nabla f_{i_k}) \leq f_{i_k}(w_k) - c\alpha_k \|\nabla f_{i_k}(w_k)\|^2,$$

  for some constant $c > 0$.

- Without interpolation this does not work (satisfied by steps that are too large).



- With interpolation, can guarantee sufficient progress towards solution.

Stochastic Line Search - Theory

- Consider using the largest step-size satisfying Armijo condition on $[0, \alpha_{\max}]$.
  - Under interpolation and strong-convexity, $c = 1/2$ and $\alpha_{\max}$ sufficiently large gives

$$\mathbb{E}\left[\|w_k - w_*\|^2\right] = \left(1 - \frac{\mu}{L_{\max}}\right)^k \|w_0 - w_*\|^2.$$

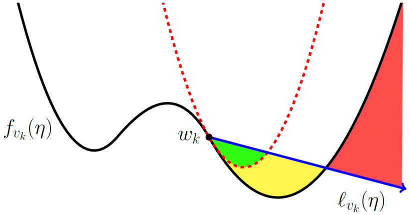
  - This is the same rate we achieve when we know the smoothness constant.
    - Under interpolation or under the WGC with the worst ρ.
  - For convex objectives we obtain an $O(1/k)$ rate.
  - For non-convex objectives we obtain the $O(1/k)$ rate if $\alpha_{\max}$ is small enough.

- In practice, we can use a backtracking line search.
  1. Start with some initial step size.
  2. Test the Armijo condition (requires an extra forward pass for neural networks).
  3. If condition is not satisfied, decrease step size and go to 2.

Superiority of Line Search over Theoretical Step Sizes

- The line search guarantees same rate as when we know smoothness constant.
  - But this is in the worst case.

- We expect the line-search to converge faster in practice.



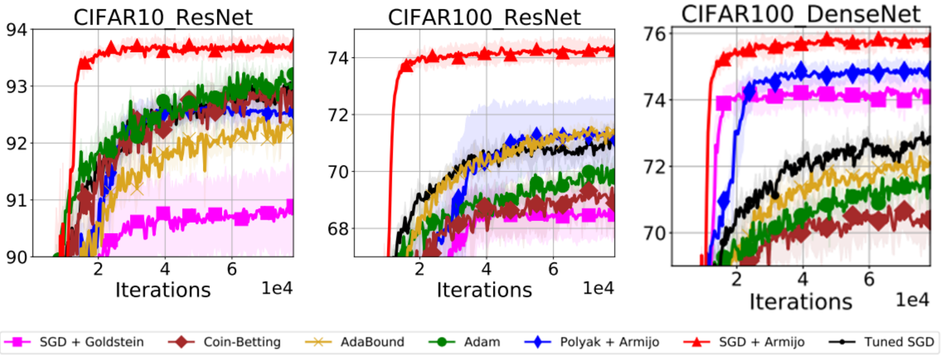
- Red dotted line is bound obtained with known smoothness for an f_i.
  - Using $\alpha_k = 1/L_{\max}$ moves to minimizer within green region.
- Armijo accepts step sizes in the yellow region (blue line is gradient of an f_i).
  - Armijo allows larger step sizes that decrease the function by a larger amount.

Stochastic Line Search - Practice

- In our experiments:
  - We used $c = 0.1$ in the Armijo condition.
  - We multiply the step size by 0.8 if the Armijo condition fails.
  - We increase the step size between iterations.
    - Specifically, we initialize the line search with $\max\{10, \alpha_{k-1} 2^{(\text{ratio of training data used})}\}$.
- With these choices, median number of times we test Armijo condition was 1.
  - Running this algorithm has similar cost to trying 2 fixed step sizes.

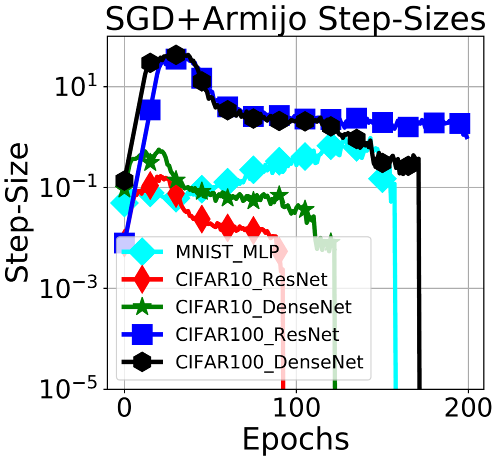
Experimental Results with Stochastic Line Search

- We did a variety of experiments, including training CNNs on standard problems.
  - Better than fixed step sizes, adaptive methods, alternate adaptive step sizes.



Experimental Results with Stochastic Line Search

- Step sizes over time under line search for different datasets.



Stochastic Line Search - Discussion

- The same line search can be used for different types of functions.
  - Strongly-convex, PL, or convex. (And bounded below under restriction of α_{max}.)
  - Adaptivity to problem difficulty.

- We ran synthetic experiments conrolling degree of over-parameterization.
  - With over-parameterization, the stochastic line search works great.
  - If close to over-parameterized, line search still works really well.
    - Theory can be modified to handle case of being close to over-parameterized.
  - If far from over-parameterized, line search catastrophically fails.

- The stochastic line-search has now been used in other algorithms.
  - Meng et al. [2020] use it to set the step size in a second-order method.
  - Vaswani et al. [2020] show that it speeds up AdaGrad and Adam empirically.

Stochastic Line Search - Concurrent Methods from October 2018

- Berrada et al. [2019] proposed an step size strategy.
  - Requires knowing f^* but step size has closed form (no backtracking).
  - Related to the stochastic Polyak step size later analyzed by Loizou et al. [2020].
    - We have found that the stochastic line search typically performs better in practice.

- Asi and Duchi [2019] considered using better models than SGD.
  - Proximal-point iterations or using truncated linear approximations.
    - For potentially non-smooth problems.
  - Obtain adaptivity to problem and fast convergence for any step size.
    - Though constants depend on the chosen step size.

- Comparison of 14 methods across 9 datasets:
  - `https://github.com/haven-ai/optimization-toolkit#Leaderboard`

Problems with Current Over-Parameterization Optimization Theory

- Line search experiments were done with batch normalization.
  - This is not covered by the theory.
  - Armijo still seems effective but gap is not as large.

- Line search is not as effective for LSTMs or transformers.
  - Adam seems to have an advantage here.
  - Theoretical and practical details to be worked out.

- Some deep learning losses like in GANs do not fit over-parameterized regime.

  [Chavdarova et al., 2019]

- Theory is still incomplete for non-convex functions:
  - Interpolation/WGC not sufficient for SGD to converge for non-convex.
    - Non-convex results rely on PL or SGC.
  - Line-search is not sufficient for convergence on non-convex.
    - Non-convex results require $\alpha_{\text{max}} = O(1/L)$.

Single-Slide Summary of this Talk

- For over-parameterized models, you need to re-think how optimization works!

  1. Stochastic gradient descent converges faster for over-parameterized models.
     - May help explain the empirical success of constant step sizes in practice.
     - May help explain why it has been so difficult to develop faster algorithms.
  2. We can design faster stochastic algorithms for over-parameterized models.
     - Over-parameterization allows Nesterov acceleration and second-order methods.
     - Over-parameterization allows better sampling schemes and tighter regret bounds.
  3. We can design stochastic algorithms that are easier to use:
     - Algorithms that do not depend on problem-dependent constants.
     - Algorithms that adapt to the difficulty of the problem.
- Thank you for the invite and taking the time to listen to the end.