535520: Optimization Algorithms

Lecture 13 — Dual Ascent and ADMM

Ping-Chun Hsieh (謝秉均)

December 9, 2024

This Lecture

1. Dual Ascent

2. Alternating Direction Method of Multipliers (ADMM)

- Reading Material:
  - Chapter 9 of Stephen Boyd's textbook "Convex Optimization"
  - S. Boyd et al., "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers," 2011
  - Chapter 15 of Amir Beck's textbook "First-Order Methods in Optimization"
  - Chapter 8 of Ian Goodfellow's textbook "Deep Learning"
  - Part of the slides are adapted from Prof. Ryan Tibshirani's lecture slides
  - Part of the slides on NNs are adapted from Prof. Suvrit Sra's lecture slides

Some Historical Accounts on ADMM

A DUAL ALGORITHM FOR THE SOLUTION OF NONLINEAR VARIATIONAL PROBLEMS VIA FINITE ELEMENT APPROXIMATION

DANIEL GABAY
Centre National de la Recherche Scientifique, Laboratoire d'Analyse Numérique, L. A. 189, 4, Place Jussieu, 75230 Paris Cedex 05, France

BERTRAND MERCIER*
IRIA, Laboria, Domaine de Voluceau, 78150, Le Chesnay, France

Communicated by R. Glowinski

(*Received May*, 1975)

Abstract—For variational problems of the form

$$\inf_{v \in V} \{f(Av) + g(v)\},$$

we propose a dual method which decouples the difficulties relative to the functionals f and g from the possible ill-conditioning effects of the linear operator A.

The approach is based on the use of an Augmented Lagrangian functional and leads to an efficient and simply implementable algorithm. We study also the finite element approximation of such problems, compatible with the use of our algorithm. The method is finally applied to solve several problems of continuum mechanics.

ADMM was first proposed by Gabay and Mercier back in 1976

Bertrand Mercier
(Professor @ CEA/INSTN in France)

INSTN = Institute for Nuclear Science and Technology

Some Historical Accounts on ADMM

Foundations and Trends® in
Machine Learning
Vol. 3, No. 1 (2010) 1–122

DOI: 10.1561/2200000016

now
the essence of knowledge

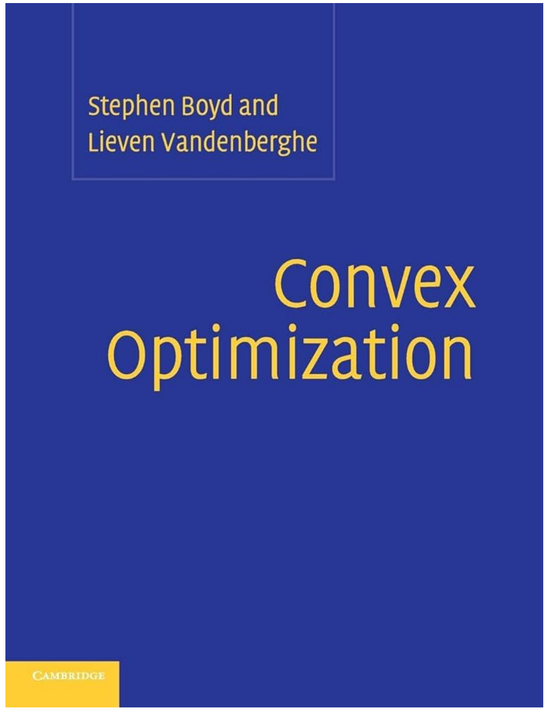
Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers

Stephen Boyd[1], Neal Parikh[2], Eric Chu[3] Borja Peleato[4] and Jonathan Eckstein[5]

[1] *Electrical Engineering Department, Stanford University, Stanford, CA 94305, USA, boyd@stanford.edu*
[2] *Computer Science Department, Stanford University, Stanford, CA 94305, USA, npparikh@cs.stanford.edu*
[3] *Electrical Engineering Department, Stanford University, Stanford, CA 94305, USA, echu508@stanford.edu*
[4] *Electrical Engineering Department, Stanford University, Stanford, CA 94305, USA, peleato@stanford.edu*
[5] *Management Science and Information Systems Department and RUTCOR, Rutgers University, Piscataway, NJ 08854, USA, jeckstei@rci.rutgers.edu*

Cited for almost 24,000 times

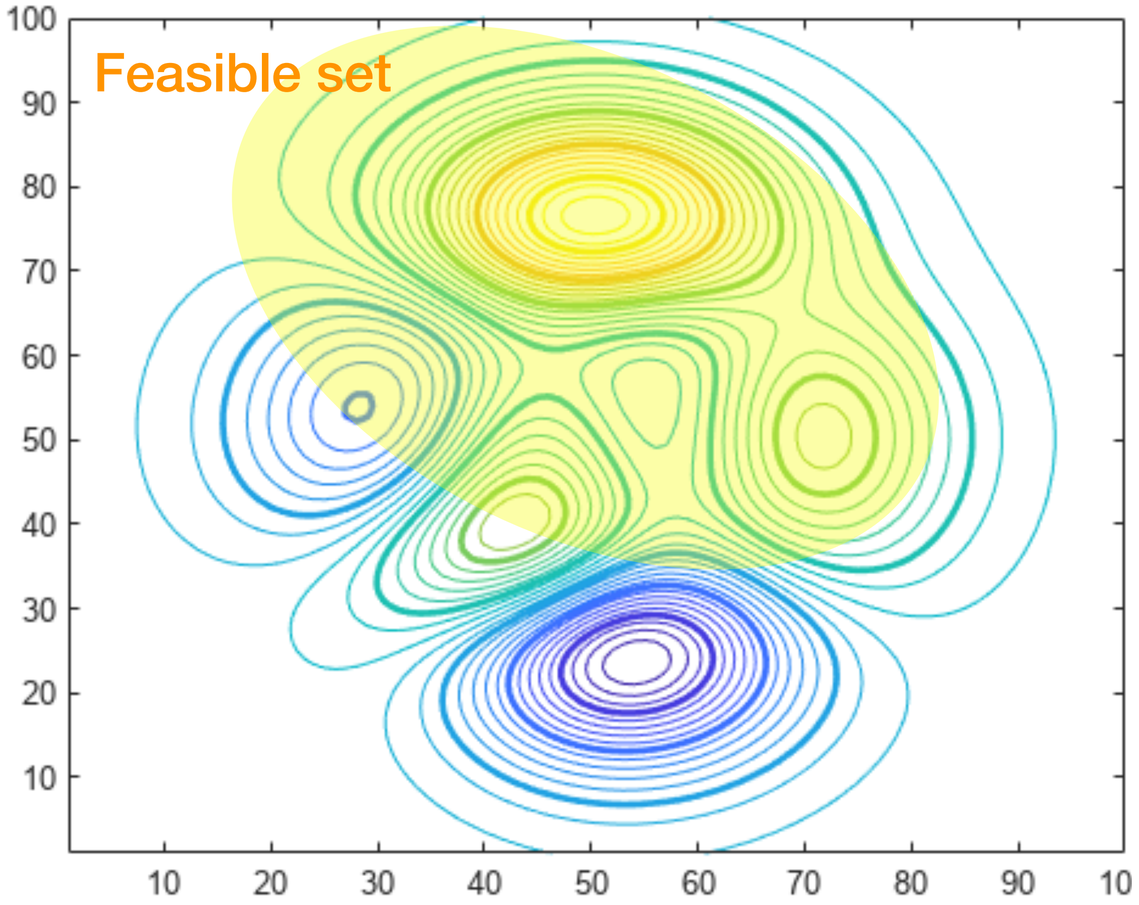
Stephen Boyd
(Professor @ Stanford)

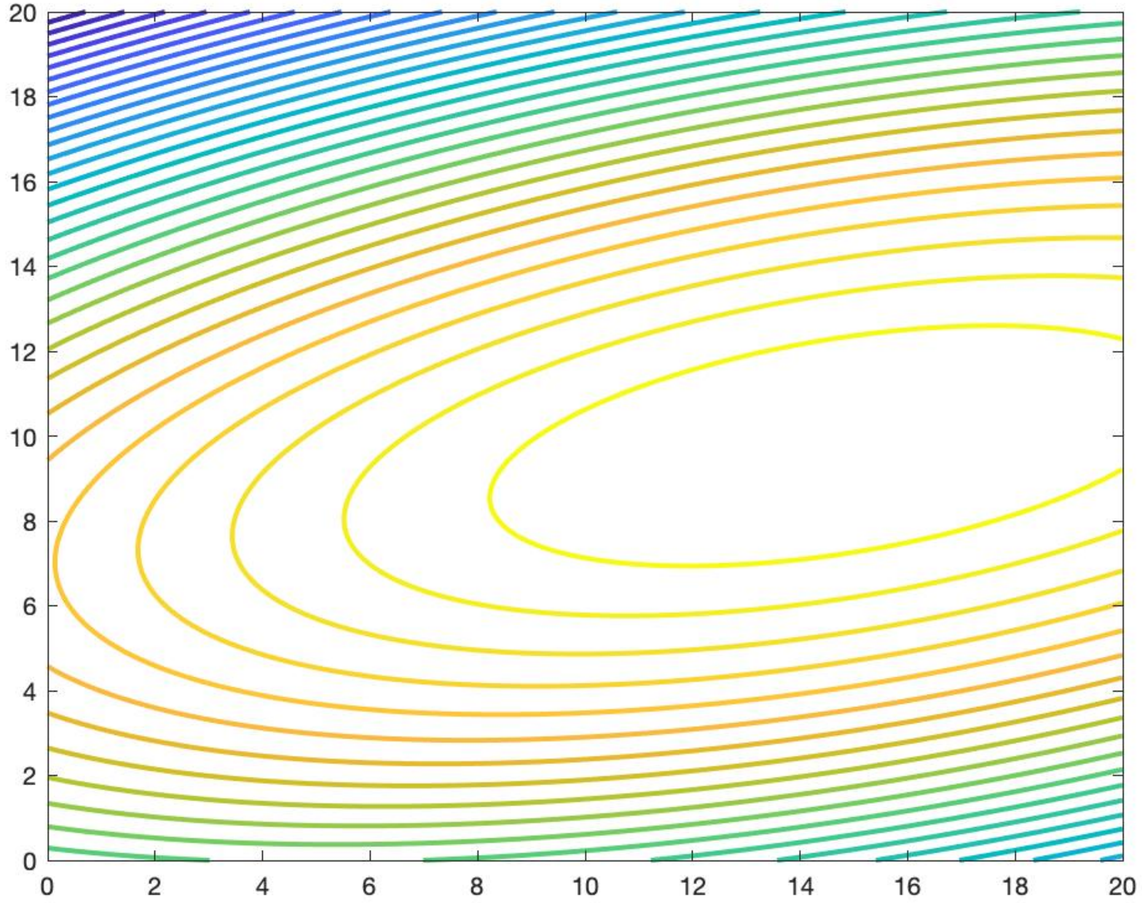


Review: Dual Ascent:

Primal: $\min_x f(x)$ s.t. $g(x) \leq 0$

Dual: $\max_{\lambda \geq 0} g(\lambda)$





Suppose strong duality holds, then?

Review: Conjugate Functions (or Legendre Transform)

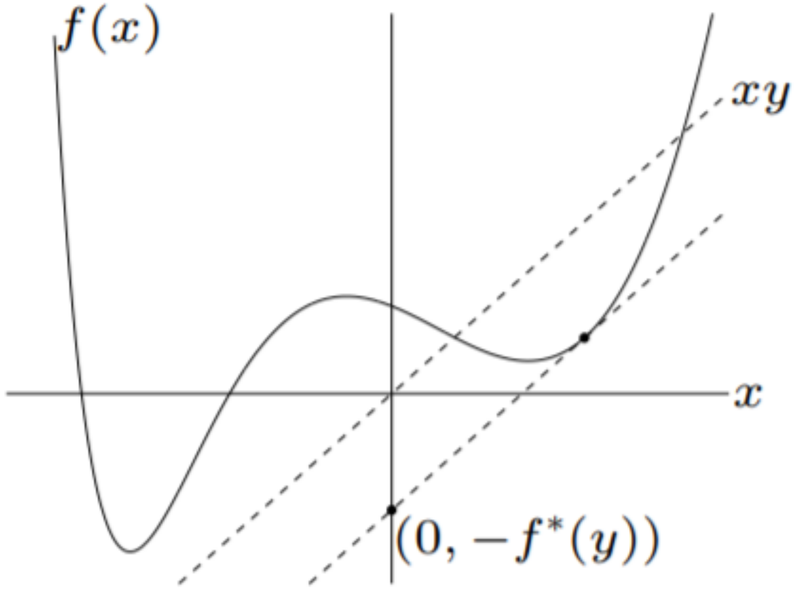
- Let $f : X \to \mathbb{R}$ be a real function (not necessarily differentiable)
- The convex conjugate $f^* : \mathbb{R}^n \to \mathbb{R}$ is defined as

$$f^*(y) := \max_{x \in X} \left(y^\top x - f(x) \right) = - \min_{x \in X} \left(f(x) - y^\top x \right)$$

Remark: Conjugates appear frequently in dual problems since $-f^*(y) = \min_x \left(f(x) - y^\top x \right)$

Remark: Conjugate of a differentiable function is also called the "Legendre-Fenchel transformation"

Question: Why is it called "convex" conjugate?



(Figure Source: Stephen Boyd's textbook)

Examples: Convex Conjugates

① $f(x) = ax + b, \quad a, b \in \mathbb{R}$

$$f^*(y) = \max_{x \in X} \left(y^T x - f(x) \right) = \begin{cases} \quad , & \text{if } y = a \\ \quad , & \text{otherwise} \end{cases}$$

② $f(x) = a^T x + b, \quad x \in \mathbb{R}^d, \; b \in \mathbb{R}, \; a \in \mathbb{R}^d$

$$f^*(y) = \max_{x \in X} \left(y^T x - f(x) \right) = \begin{cases} \quad , & \text{if } y = a \\ \quad , & \text{otherwise} \end{cases}$$

Nice Properties of Conjugate Functions

Property 1: If f is differentiable and convex, then $\bar{x}$ is a maximizer of $y^\top x - f(x)$ if and only if $y = \nabla f(\bar{x})$

Property 2: If f is μ-strongly convex, then f^* is $\frac{1}{\mu}$-smooth

Property 3: If f is convex and closed (i.e., the epigraph of f is a closed set), the $f^{**} = f$

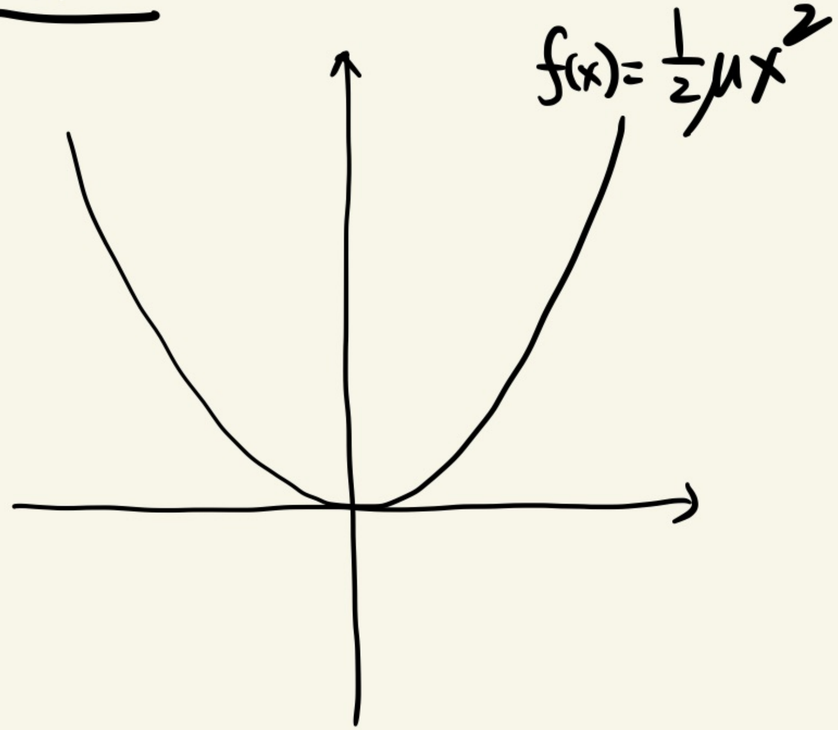
(Proof: Please see Section 3.3 of Stephen Boyd's textbook)

Property 4: If f is convex and closed (i.e., the epigraph of f is a closed set), then $x = \nabla f^*\big(\nabla f(x)\big)$

Smoothness of f and f^*

Lemma: If f is a μ-strongly convex function, then f^* is $\frac{1}{\mu}$-smooth.

Intuition:



$$f^*(y) = \max_x \left(y^T x - f(x) \right)$$

$$= \max_x \left(yx - \frac{1}{2}\mu x^2 \right)$$

Smoothness of f and f^*

Lemma: If f is a μ-strongly convex function, then f^* is $\frac{1}{\mu}$-smooth.

Proof: **Step1** Since f is μ-strongly convex, then $f(z) \geq f(x) + \frac{\mu}{2}\|z-x\|^2$, for all x, z

Step2 Let $x_u = \nabla f^*(u)$, $x_v = \nabla f^*(v)$

$$f(x_v) - u^T x_v \geq f(x_u) - u^T x_u + \frac{\mu}{2}\|x_u - x_v\|^2$$

$$+) \quad f(x_u) - v^T x_u \geq f(x_v) - v^T x_v + \frac{\mu}{2}\|x_u - x_v\|^2$$

$$(u-v)^T(x_u - x_v) \geq \mu \cdot \|x_u - x_v\|^2$$

$$\Downarrow \text{(why?)}$$

$$\|x_u - x_v\| \leq \frac{1}{\mu}\|u - v\|$$

A Motivating Example of Dual Ascent

Let $f(x)$ be a convex function

$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{subject to} \quad Ax = b \qquad (A \in \mathbb{R}^{m \times d},\ b \in \mathbb{R}^d)$$

- Lagrangian: $L(x, \lambda) = f(x) + \lambda^T (Ax - b)$
- Dual function: $g(\lambda) = \min_x L(x, \lambda) = -f^*(-A^T\lambda) - b^T\lambda$
- Dual Problem: $\max_{\lambda \in \mathbb{R}^m} g(\lambda)$

Question: How to achieve "ascent" for the dual problem?

Recall from Lecture 3: Lagrangian Optimality Conditions (LOC)

Theorem (LOC):

Suppose strong duality holds and (x^*, λ^*) exists. Then, we have

$$x^* \in \arg\min_{x \in X} \mathscr{L}(x, \lambda^*)$$

- **Comparison**: Optimality condition FONC-C from Lecture 1

$$\nabla f(x^*)^\top (x - x^*) \geq 0, \quad \forall x \in A \subseteq X$$

(A is the feasible set)

A Motivating Example of Dual Ascent

$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{subject to} \quad Ax = b \quad (A \in \mathbb{R}^{m \times d}, b \in \mathbb{R}^d)$$

Dual Ascent: Let's apply gradient ascent for the dual problem

$$x_{t+1} := \operatorname{argmin}_x \mathcal{L}(x, \lambda_t) \quad \text{(minimization step)}$$

$$\lambda_{t+1} := \lambda_t + \alpha_t \cdot \nabla_\lambda g(\lambda_t) \quad \text{(dual variable update)}$$

In this example:

$$\nabla_\lambda g(\lambda_t) = A x_{t+1} - b \quad \text{(why?)}$$

Convergence of Dual Ascent?

Question: Under dual ascent, does λ_t converge to λ^*? If so, under what conditions? What is the convergence rate (in terms of $g(\lambda^*) - g(\lambda_t)$)?

Dual Decomposition

Suppose f is "separable" in the sense that

$$\min \; f(x) = f_1(x^{(1)}) + \cdots + f_M(x^{(M)}), \qquad x = (x^{(1)}, \cdots, x^{(M)})$$

$$\text{subject to} \quad Ax = b$$

- $L(x, \lambda)$ is also separable in x:

$$L(x, \lambda) = L_1(x^{(1)}, \lambda) + \cdots + L_M(x^{(M)}, \lambda) - \lambda^T b, \quad \text{where } L_i(x_i^{(i)}, \lambda) = f_i(x^{(i)}) + \lambda^T A_i x^{(i)}$$

- "Minimization Step" can be done in parallel:

$$x_{t+1}^{(i)} = \underset{x^{(i)} \in \mathbb{R}^{d_i}}{\operatorname{argmin}} \; L_i(x^{(i)}, \lambda_t)$$

Dual Decomposition Algorithm

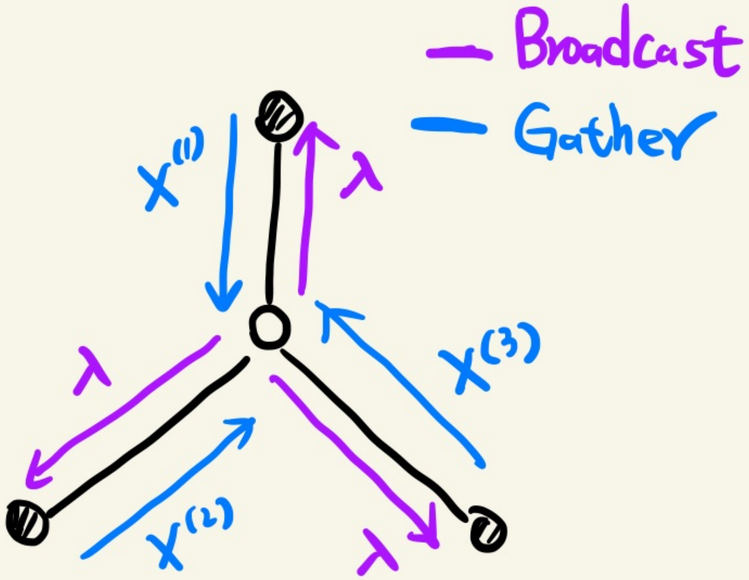
Repeat the following two steps:

Step1 (Minimization in parallel)

$$X_{t+1}^{(i)} = \underset{x^{(i)}}{\operatorname{argmin}}\ L(x^{(i)}, \lambda_t) = \underset{x^{(i)}}{\operatorname{argmin}} \left(f_i(x^{(i)}) + \lambda_t^T A_i x^{(i)} \right)$$

Step2 (Dual variable update)

$$\lambda_{t+1} = \lambda_t + \alpha_t \left(\sum_{i=1}^{M} A_i x_t^{(i)} - b \right)$$



A Quick Remark: How About Dual Ascent for Inequality Constraints?

Dual Ascent for Inequality Constraints

Let $f(x)$ be a convex function

$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{subject to} \quad Ax \leq b \qquad (A \in \mathbb{R}^{m \times d},\ b \in \mathbb{R}^d)$$

- Lagrangian: $L(x, \lambda) = f(x) + \lambda^T (Ax - b)$
- Dual function: $g(\lambda) = \min_x L(x, \lambda) = -f^*(-A^T\lambda) - b^T\lambda$
- Dual Problem: $\max_{\lambda \geq 0} g(\lambda)$

Question: How to achieve "ascent" for the dual problem?

Dual Ascent for Inequality Constraints

Let $f(x)$ be a convex function

$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{subject to} \quad Ax \leq b \quad (A \in \mathbb{R}^{m \times d},\ b \in \mathbb{R}^d)$$

Dual Ascent: Let's apply gradient ascent for the dual problem

$$x_{t+1} := \operatorname*{argmin}_x \mathcal{L}(x, \lambda_t) \qquad \text{(minimization step)}$$

$$\lambda_{t+1} := \underbrace{\left[\lambda_t + \alpha_t \cdot \nabla_\lambda g(\lambda_t)\right]^+}_{\text{Projected Gradient Ascent.}} \qquad \text{(dual variable update)}$$

Question:

Convergence rate?

In this example: $\nabla_\lambda g(\lambda_t) = A x_{t+1} - b$

Recall: Convergence of PGD for Convex and Smooth Problems

Theorem (Convergence of PGD):

Let f be convex and L-smooth. Under PGD with constant step sizes $\eta = 1/L$, we have

$$f(x_t) - f(x^*) \leqslant \frac{3L\|x_0 - x^*\|^2 + \big(f(x_0) - f(x^*)\big)}{t+1}, \quad \forall t \in \mathbb{N}$$

- Remark: PGD achieves $O(1/t)$ rate as GD for unconstrained problems

Recall: Convergence of PGD under Strong Convexity

Theorem (Convergence of PGD):

Let f be μ-strongly convex and L-smooth. Under PGD with constant step sizes $\eta = 1/L$, we have

$$\|x_t - x^*\|^2 \leqslant \left(1 - \frac{\mu}{L}\right)^t \cdot \|x_0 - x^*\|^2$$

Summary: Pros and Cons of Dual Ascent

Pro 1: Dual ascent can lead to a decentralized algorithm

Con 1: Convergence require strong assumptions (e.g., strong convexity) and can be slow (even under strong convexity of f)

Con 2: The "minimization step" of dual ascent could diverge! (e.g., $f(x) = Cx + d$)

$$x_{t+1} = \arg\min_x L(x, \lambda_t)$$
$$\lambda_{t+1} = \lambda_t + \alpha_t \nabla g(\lambda_t)$$

Augmented Lagrangian
(aka Methods of Multipliers)

Augmented Lagrangian Method

Primal

$$\min_x \; f(x) \quad \text{subject to} \quad Ax = b$$

$\Longrightarrow$

Equivalent Primal

$$\min_x \; f(x) + \frac{\rho}{2}\|Ax-b\|^2 \quad \text{subject to} \quad Ax = b$$

$\frac{\rho}{2}\|Ax-b\|^2$: augmented Lagrangian

Dual Ascent

$$x_{t+1} = \operatorname{argmin}_x \underbrace{\left(f(x) + \lambda_t^T Ax + \frac{\rho}{2}\|Ax-b\|^2 \right)}_{=: L_\rho(x,\lambda_t)}$$

$$\lambda_{t+1} = \lambda_t + \underbrace{\rho}_{\text{step size}} \cdot (Ax_{t+1} - b)$$

Question: Why choosing $\alpha_t = \rho$?

Augmented Lagrangian Method: Step Size

- Optimality Conditions (suppose f is differentiable)

$$Ax^* - b = 0 \quad \text{(primal feasibility)}$$

$$\nabla_x \mathcal{L}(x, \lambda) = \nabla f(x^*) + A^T \lambda^* = 0 \quad (\nabla_x \mathcal{L}(x, \lambda) = 0)$$

- Since x_{t+1} is a minimizer of $\mathcal{L}_\rho(x, \lambda_t)$, we have

$$\begin{aligned} 0 &= \nabla_x \mathcal{L}_\rho(x_{t+1}, \lambda_t) \\ &= \nabla_x f(x_{t+1}) + A^T\big(\lambda_t + \rho(Ax_{t+1} - b)\big) \\ &= \nabla_x f(x_{t+1}) + A^T \lambda_{t+1} \end{aligned}$$

$\Rightarrow (x_{t+1}, \lambda_{t+1})$ satisfies

Pros and Cons of Augmented Lagrangian

Pro 1: Convergence under more relaxed assumptions (e.g., f is convex but not strongly convex)

Pro 2: The “minimization step” of augmented Lagrangian would not diverge under a linear objective function!

$$x_{t+1} = \arg\min_{x} L_{\rho}(x, \lambda_t)$$

Con 1: Decomposability no longer holds! (Why?)

Alternating Direction Method of Multipliers (ADMM)

(Best of both worlds: (1) decomposability of dual ascent and (2) convergence properties of the method of multipliers)

A Motivating Problem for ADMM: Global Consensus

Global Variable Consensus Optimization

$$\min_{x \in \mathbb{R}^n} f(x) = \sum_{i=1}^{N} f_i(x).$$

Example

- In typical ERM problems with N data samples

$$L(\theta) = \frac{1}{N} \sum_{i=1}^{N} \ell_i(\theta)$$

Reformulated as

Global Consensus problem

$$\min f(x) = \sum_{i=1}^{N} f_i(x_i), \quad X = (x_1, x_2, \cdots, x_N)$$

Subject to $x_i - z = 0$, for all i

Alternating Direction Method of Multipliers (ADMM)

Consider the problem (convex functions)

$$\min_{x,z} \quad f(x)+h(z)$$
$$\text{subject to} \quad Ax+Bz=c$$

(no assumption on A, B, c)

$\Rightarrow$ Equivalent Problem ($\rho>0$)

$$\min \quad f(x)+h(z)+\frac{\rho}{2}\|Ax+Bz-c\|^2$$
$$\text{subject to} \quad Ax+Bz=c$$

Define "augmented Lagrangian" as

$$L_\rho(x,z,\lambda) = f(x)+h(z)+\lambda^T(Ax+Bz-c)+\frac{\rho}{2}\|Ax+Bz-c\|^2$$

- ADMM:

$$\begin{cases} x_{t+1} = \underset{x}{\operatorname{argmin}}\ L_\rho(x, z_t, \lambda_t) \\ z_{t+1} = \underset{z}{\operatorname{argmin}}\ L_\rho(x_t, z, \lambda_t) \\ \lambda_{t+1} = \lambda_t + \rho(Ax_t+Bz_t-c) \end{cases}$$

Split minimization! (rather than joint minimization in the vanilla augment Lagrangian)

Convergence Guarantees of ADMM

Theorem: Suppose the following assumptions hold:

(1) Both $f(x)$ and $h(x)$ are convex and closed

(2) The unaugmented Lagrangian $L_0(x,z,\lambda)$ has a saddle point, i.e. there exists (x^*,z^*,λ^*)

Such that $L_0(x^*,z^*,\lambda) \le L_0(x^*,z^*,\lambda^*) \le L_0(x,z,\lambda^*)$, for all x,z,λ

⇓ this implies that (x^*,z^*) is a primal solution and λ^* is dual solution

Then, we have

(i) (Residual convergence) $Ax_t + Bz_t - c \to 0$, as $t \to \infty$

(ii) (Objective convergence) $f(x_t) + g(z_t) \to p^*$, as $t \to \infty$

(iii) (Dual convergence) $\lambda_t \to \lambda^*$, as $t \to \infty$

Wei Deng and Wotao Yin, "On the global and linear convergence of the generalized alternating direction method of multipliers," Journal of Scientific Computing, 2016

Scaled Form of ADMM

Goal: Combine the linear and quadratic terms.

- Let $r = Ax + Bz - c$
- $$\lambda^T r + \frac{\rho}{2}\|r\|^2 = \frac{\rho}{2}\|r + \frac{1}{\rho}\lambda\|^2 - \frac{1}{2\rho}\|\lambda\|^2$$

(scaled dual variable)

$$= \frac{\rho}{2}\|r + u\|^2 - \frac{\rho}{2}\|u\|^2, \text{ where } u = \frac{1}{\rho}\lambda$$

Scaled ADMM

$$x_{t+1} = \underset{x}{\operatorname{argmin}} \left(f(x) + \frac{\rho}{2}\|Ax + Bz_t - c + u_t\|^2 \right)$$

$$z_{t+1} = \underset{z}{\operatorname{argmin}} \left(h(z) + \frac{\rho}{2}\|Ax_{t+1} + Bz - c + u_t\|^2 \right)$$

$$u_{t+1} = u_t + Ax_{t+1} + Bz_{t+1} - c$$

Example of ADMM: Alternating Projections

Suppose we'd like to find a point in intersection of convex sets, e.g., $C_1, C_2 \subseteq \mathbb{R}^d$

$$\min_x \; I_{C_1}(x) + I_{C_2}(x)$$

ADMM $\min_{x,z} \; I_{C_1}(x) + I_{C_2}(z)$, subject to $x - z = 0$

Then, in each iteration, ADMM shall take the following steps

$$x_{t+1} = \operatorname{argmin}_x \mathcal{L}_\rho(x, z_t, \lambda_t) = \Pi_{C_1}(z_t - \lambda_t)$$

$$z_{t+1} = \operatorname{argmin}_z \mathcal{L}_\rho(x_{t+1}, z, \lambda_t) = \Pi_{C_2}(x_{t+1} + \lambda_t)$$

$$\lambda_{t+1} = \lambda_t + x_{t+1} - z_{t+1}$$

ADMM for Global Consensus Problem

$$\min \quad f(x) = \sum_{i=1}^{N} f_i(x^{(i)})$$

$$\text{subject to} \quad x^{(i)} - z = 0, \quad \text{for all } i = 1, \cdots, N$$

ADMM: Augmented Lagrangian $L_\rho(x_1, \cdots, x_N, z, \lambda) = \sum_{i=1}^{N} f_i(x^{(i)}) + \lambda_i^T (x^{(i)} - z) + \frac{\rho}{2} \| x^{(i)} - z \|^2$

$$x_{t+1}^{(i)} = \underset{x^{(i)}}{\operatorname{argmin}} \left(f_i(x^{(i)}) + \lambda_t^{(i)T} (x^{(i)} - z_t) + \frac{\rho}{2} \| x^{(i)} - z_t \|^2 \right)$$

$$z_{t+1} = \frac{1}{N} \sum_{i=1}^{N} \left(x_{t+1}^{(i)} + \frac{1}{\rho} \lambda_t^{(i)} \right)$$

$$\lambda_{t+1}^{(i)} = \lambda_t^{(i)} + \rho \left(x_{t+1}^{(i)} - z_{t+1} \right).$$

Numerical Example: Distributed Regularized Logistic Regression

Label (-1, or +1)

$$\min_{w,v} \quad \sum_{i=1}^{m} \log(1 + \exp(-b_i(a_i^\top w + v))) + \eta \|w\|_1$$

Feature vector

($m = 10^6$ and divided into 100 subgroups)

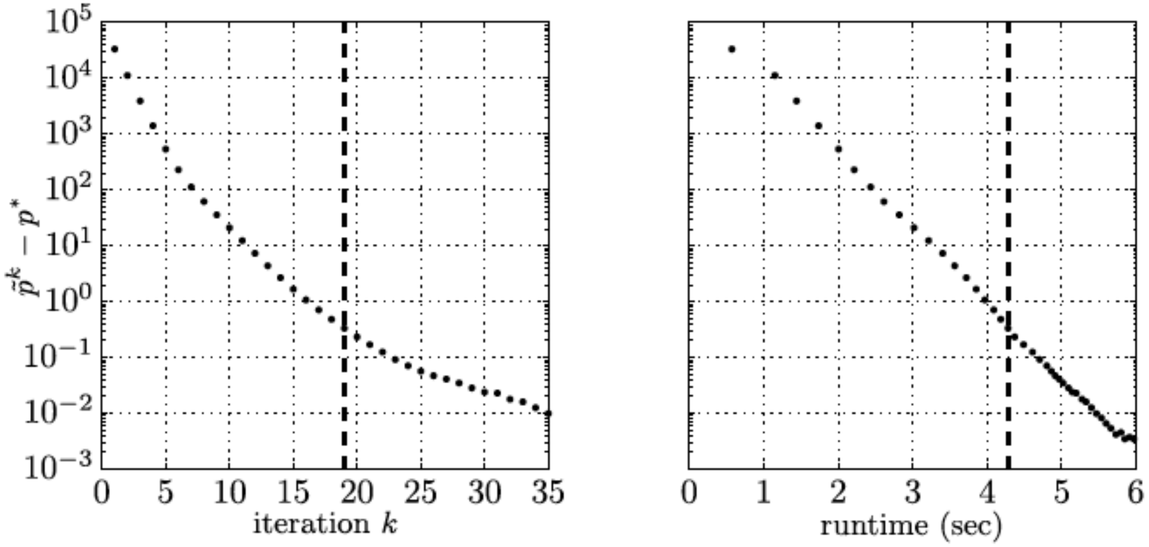


Fig. 11.5. *Left.* Objective suboptimality of distributed ℓ_1 regularized logistic regression versus iteration. *Right.* Progress versus elapsed time. The stopping criterion is satisfied at iteration 19, indicated by the vertical dashed line.

Sub-optimality gap

Primal and dual residual

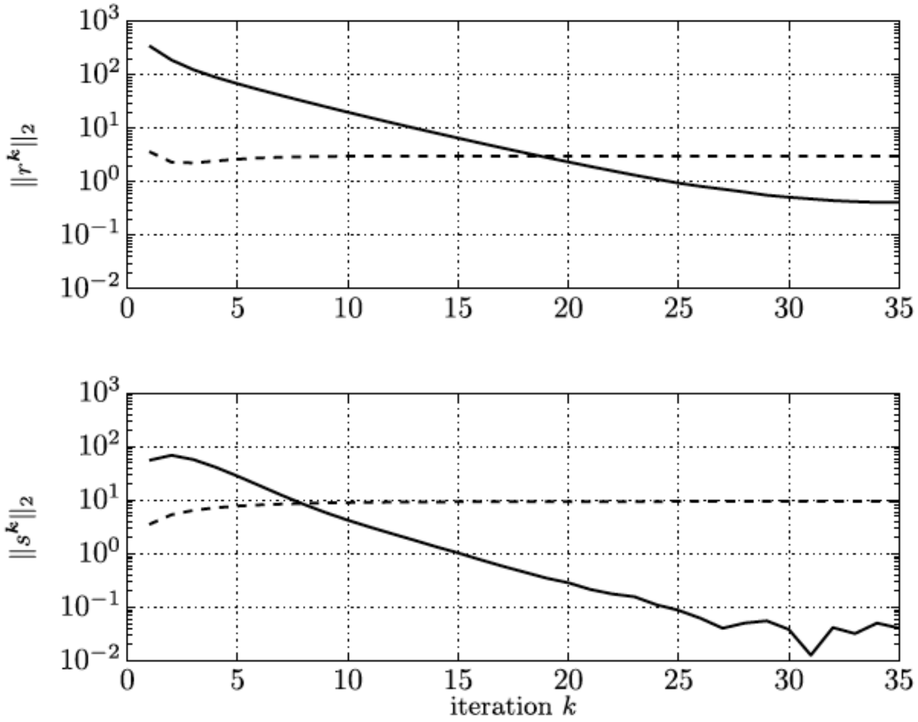


Fig. 11.4. Progress of primal and dual residual norm for distributed ℓ_1 regularized logistic regression problem. The dashed lines show ϵ^{pri} (top) and ϵ^{dual} (bottom).

(Example Source: Chapter 11 of Stephen Boyd's paper)

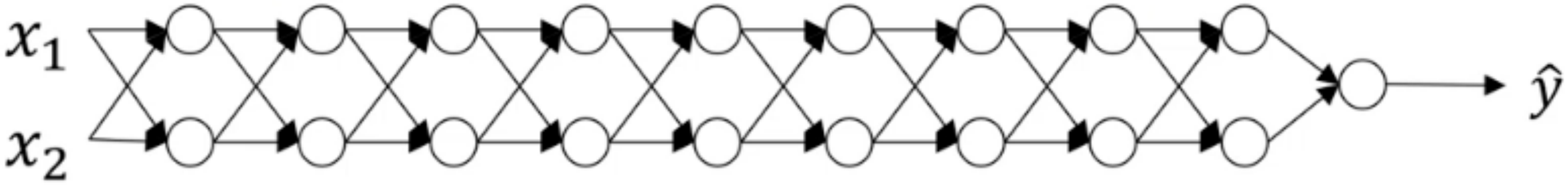
Optimization Techniques for Neural Networks

Some Popular Optimization Techniques for Training NNs

- Clipped gradient
- Adaptive gradient and AdaGrad (Duchi et al., 2011)
- ADAM (Kingma and Ba, 2014)
- RMSProp (Tieleman and Hinton, 2012)
- Batch Normalization
- And more!

Exploding / Vanishing Gradients in NNs

- A motivating example: Fully-connected linear networks with hidden size = 2



(Example Source: Andrew Ng's DL lecture)

Clipped GD vs Normalized GD

- Clipped GD (C-GD)

$$x_{t+1} = x_t - \min\left(\eta_c, \frac{\gamma\eta_c}{\|\nabla f(x_t)\|}\right)\nabla f(x_t)$$

- Normalized GD (N-GD)

$$x_{t+1} = x_t - \frac{\eta_n}{\|\nabla f(x_t)\| + \beta}\nabla f(x_t)$$

Remark: C-GD and N-GD are equivalent up to a constant in step size (by setting $\gamma\eta_c = \eta_n$ and $\eta_c = \eta_n/\beta$)

1. How about the case where $\|\nabla f(x_t)\|$ is large?

2. What if $\|\nabla f(x_t)\|$ is small?

Theoretical Justification Behind Clipped GD

WHY GRADIENT CLIPPING ACCELERATES TRAINING: A THEORETICAL JUSTIFICATION FOR ADAPTIVITY

Jingzhao Zhang, Tianxing He, Suvrit Sra & Ali Jadbabaie
Massachusetts Institute of Technology
Cambridge, MA 02139, USA
{jzhzhang, tianxing, suvrit, jadbabai}@mit.edu

ABSTRACT

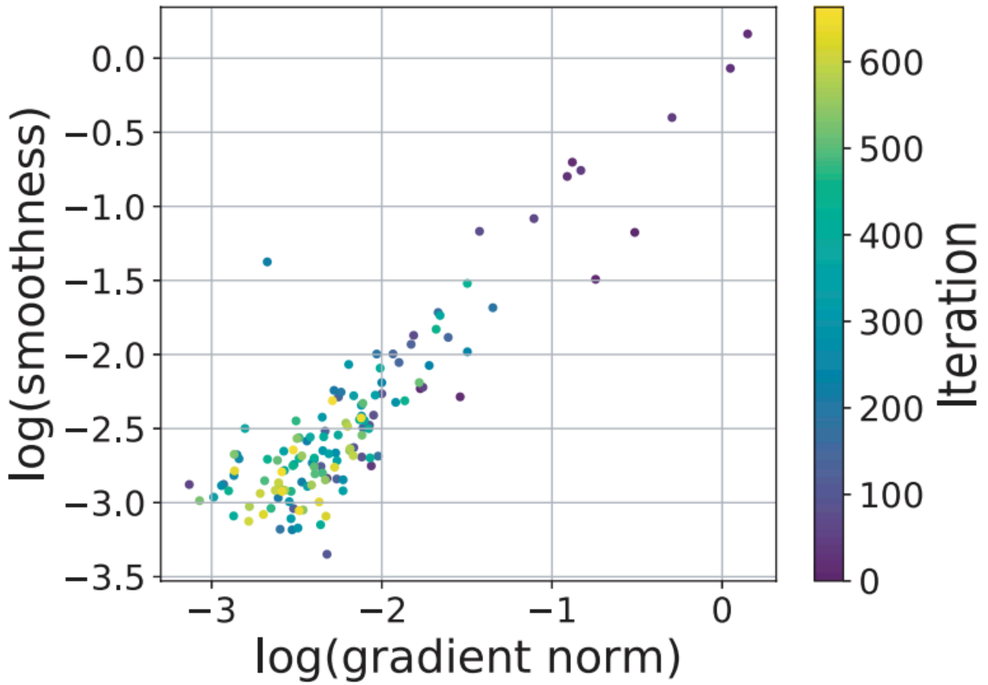
We provide a theoretical explanation for the effectiveness of gradient clipping in training deep neural networks. The key ingredient is a new smoothness condition derived from practical neural network training examples. We observe that gradient smoothness, a concept central to the analysis of first-order optimization algorithms that is often assumed to be a constant, demonstrates significant variability along the training trajectory of deep neural networks. Further, this smoothness positively correlates with the gradient norm, and contrary to standard assumptions in the literature, it can grow with the norm of the gradient. These empirical observations limit the applicability of existing theoretical analyses of algorithms that rely on a fixed bound on smoothness. These observations motivate us to introduce a novel relaxation of gradient smoothness that is weaker than the commonly used Lipschitz smoothness assumption. Under the new condition, we prove that two popular methods, namely, *gradient clipping* and *normalized gradient*, converge arbitrarily faster than gradient descent with fixed stepsize. We further explain why such adaptively scaled gradient methods can accelerate empirical convergence and verify our results empirically in popular neural network training settings.

(Zhang et al., ICLR 2020)

The main findings of this work:

- A new “gradient smoothness condition” (rather than Lipschitz gradient condition) is needed for practical neural network training examples
- Under the new condition, C-GD cab converge arbitrarily faster than gradient descent with fixed step size

Motivating Experiments



Gradient norm vs local gradient Lipschitz constant on a log-scale along the training trajectory for a variant of LSTM (Merity et al., 2018) on PTB dataset

L-smoothness:

$$\|\nabla^2 f(x)\| \leq L$$

Issues: Such a global L can be too conservative or does not exist

(L_0, L_1)-smoothness:

$$\|\nabla^2 f(x)\| \leq L_0 + L_1 \|\nabla f(x)\|$$

Example: $f(x) = x^\alpha$, for $\alpha \geq 3$

Zhang et al., "Why gradient clipping accelerates training: A theoretical justification for adaptivity," ICLR 2020

Convergence of Clipped GD

Assumption 1. The function f is lower bounded by $f^* > -\infty$.

Assumption 2. The function f is twice differentiable.

Assumption 3 ((L_0, L_1)-smoothness). The function f is (L_0, L_1)-smooth, i.e., there exist positive constants L_0 and L_1 such that $\|\nabla^2 f(x)\| \leq L_0 + L_1\|\nabla f(x)\|$—see condition (2).

Clipped GD converges at a typical rate

Theorem 3. *Let $\mathcal{F}$ denote the class of functions that satisfy Assumptions 1, 2, and 3 in set $\mathcal{S}$ defined in (3). Recall f^* is a global lower bound for function value. With $\eta_c = \frac{1}{10L_0}$, $\gamma = \min\{\frac{1}{\eta_c}, \frac{1}{10L_1\eta_c}\}$, we can prove that the iteration complexity of clipped GD (Algorithm 5) is upper bounded by*

$$\frac{20L_0(f(x_0) - f^*)}{\epsilon^2} + \frac{20\max\{1, L_1^2\}(f(x_0) - f^*)}{L_0}.$$

GD can converge slowly, especially when $L_1 M$ is large

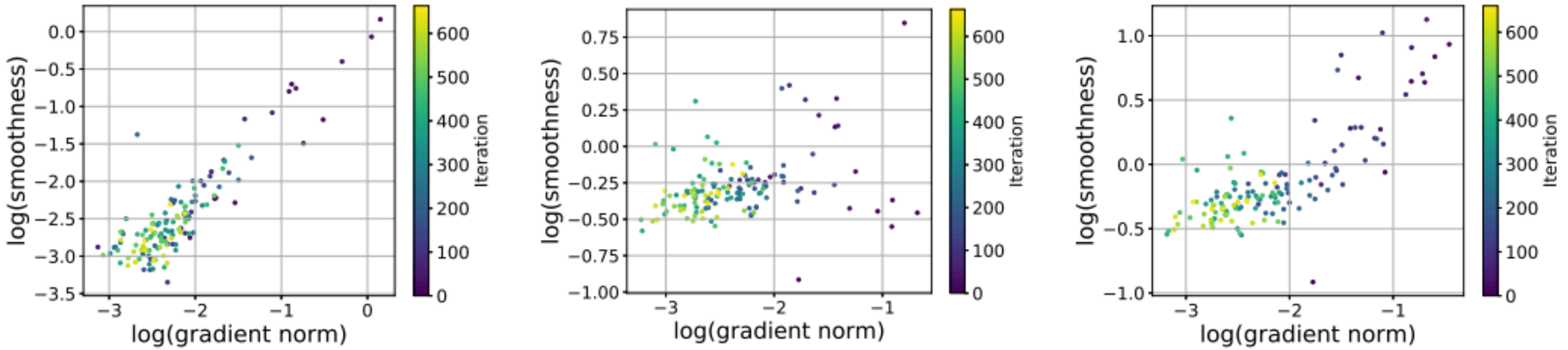
$$M := \sup\{\|\nabla f(x)\| | f(x) \leq f(x_0)\}$$

Theorem 4. *Let $\mathcal{F}$ be the class of objectives satisfying Assumptions 1, 2, 3, and 4 with fixed constants $L_0 \geq 1$, $L_1 \geq 1$, $M > 1$. The iteration complexity for the fixed-step gradient descent algorithms parameterized by step size h is at least*

$$\frac{L_1 M(f(x_0) - f^* - 5\epsilon/8)}{8\epsilon^2(\log M + 1)}.$$

Zhang et al., “Why gradient clipping accelerates training: A theoretical justification for adaptivity,” ICLR 2020

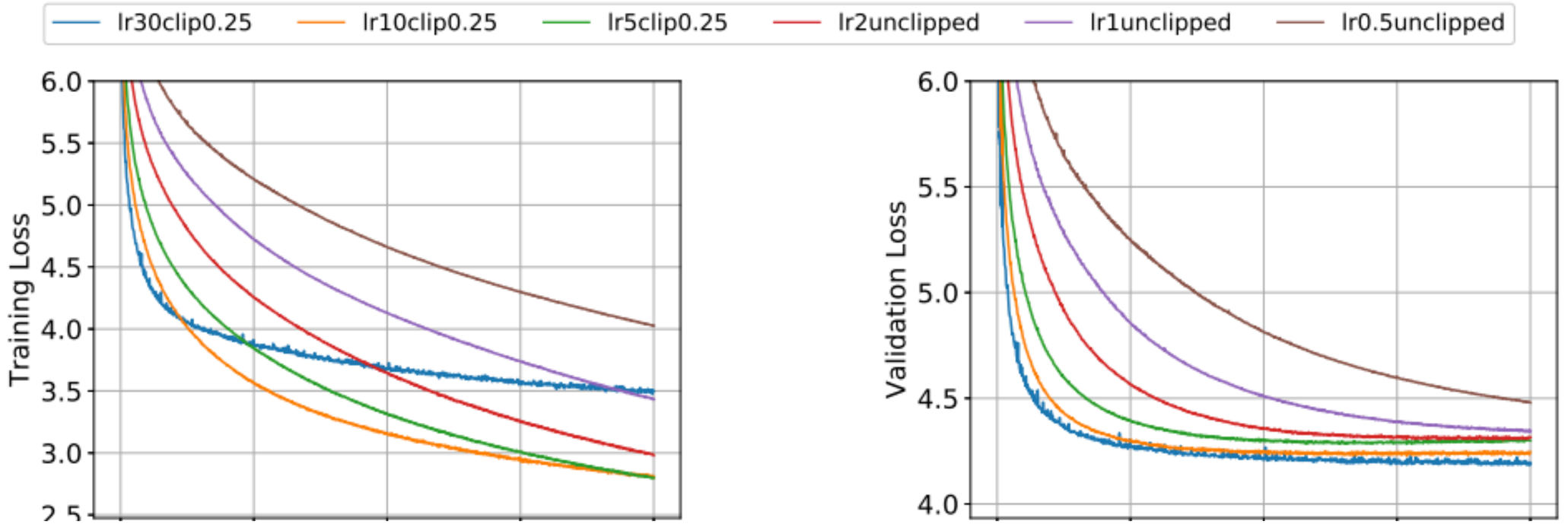
Empirical Validation



(a) Learning rate 30, with clipping. (b) Learning rate 2, without clipping. (c) Learning rate 2, with clipping.

- Clipping enables the training trajectory to stably traverse non-smooth regions
- Gradient norms and smoothness are positively correlated under large learning rates

Empirical Validation



Training loss of LSTM

Validation loss of LSTM

- Clipping indeed accelerates convergence (as typically observed)