Adam Can Converge Without Any Modification On Update Rules

Yushun Zhang , Congliang Chen, Naichen Shi, Ruoyu Sun, Zhi-Quan Luo

The Chinese University of Hong Kong, Shenzhen, China

NeurIPS 2022

Presented on Jan 11th 2023, at Google Research
Many thanks to Dr. Kingma for the invitation!

What to expect from this talk?

- **For practitioners:**
  - Is Adam a theoretically justified algorithm?
  - Shall we use it confidently?
  - When Adam does not work well, how to tune hyperparameters?

- **For theoretical researchers:**
  - Convergence & divergence phase transition.
  - Problem-dependent bound v.s. Problem-independent bound.
  - A new method to analyze stochastic non-linear dynamic system.

Overview of our results

- We prove that Adam can converge without ANY modification.
  - **Proof idea:** new observations of Adam' s non-linear dynamics under random permutations
  - **Implication:** Adam is still a theoretically justified algorithm.
    Please use it confidently!

- We provide suggestions for hyperparameter tunning.
  - **In one sentence:** First, tune up β_2. Then, try different β_1 with $\beta_1 < \sqrt{\beta_2}$
  - **Detailed suggestions:** see next page

Receipe for hyperparameter-tuning

AdaShift? AdaBound?

No need!

Change algorithm?

Adam with default setting $(\beta_1, \beta_2) = (0.9, 0.999)$ **fails** in your task

Tune up β_2.
Try 0.9995, 0.9999, 0.99995, etc

Does not work better?

Try smaller β_1.
Try 0.7, 0.5, 0.3, 0.1, etc

Adam with default setting $(\beta_1, \beta_2) = (0.9, 0.999)$ **works** in your task, but want better performance

Tune down β_2 a bit
Try 0.995, 0.99, 0.95, etc
But not too small!

Does not work better?

Seems confused? No worries. We will come back to this figure later

Contents

Contents

Motivation

- **Adam** is one of the most popular algorithms in deep learning (DL).
  (It has received more than 130,000 citations)

- **Default** choice in many DL tasks:
  - NLP, GAN, CV, GNN, RL etc.

```
optimizer = optim.Adam(net.parameters(), lr=args.lr, betas=(args.beta1, args.beta2), eps=1e-08,
                       weight_decay=args.weightdecay, amsgrad=False)
```

- However, the behavior of Adam is **poorly understood** in theory.
- We aim to close the gap between theory and practice.

A Brief Introduction: From SGD to Adam

- Consider $\min_x f(x) := \sum_{i=1}^{n} f_i(x)$.
- In DL tasks, n often stands for sample size; x denotes trainable parameters.
- In the k-th iteration: Randomly sample τ_k from $\{1,2,\ldots,n\}$

- SGD:
- $x_{k+1} = x_k - \eta_k \nabla \mathrm{f}_{\tau_k}(\mathrm{x_k})$

- SGD with momentum (SGDM):
- $m_k = (1 - \beta_1)\nabla \mathrm{f}_{\tau_k}(\mathrm{x_k}) + \beta_1 m_{k-1}$
- $x_{k+1} = x_k - \eta_k m_k$

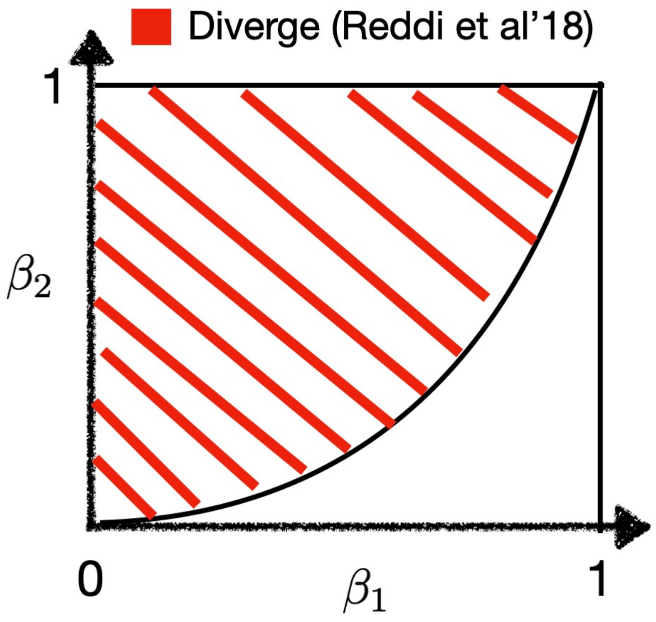
- RK: SGD & SGDM do not work well in complicated tasks (e.g., RL and NLP)

A Brief Introduction: From SGD to Adam

- Consider $\min_x f(x) := \sum_{i=1}^{n} f_i(x)$. In the k-th iteration: Randomly sample τ_k from $\{1,2,\dots,n\}$
  - Adam (Kingma and Ba'15):
  - $m_k = (1-\beta_1)\nabla f_{\tau_k}(x_k) + \beta_1 m_{k-1}$
  - $v_k = (1-\beta_2)\nabla f_{\tau_k}(x_k) \circ \nabla f_{\tau_k}(x_k) + \beta_2 v_{k-1}$
  - $x_{k+1} = x_k - \eta_k \frac{\sqrt{1-\beta_2^k}}{1-\beta_1^k} \frac{m_k}{\sqrt{v_k}}$
- β_1: Controls the 1st-order momentum m_k. Default setting: $\beta_1 = 0.9$
- β_2: Controls the 2nd-order momentum v_k. Default setting: $\beta_2 = 0.999$
- How to sample τ_k?
  - With-replacement sampling (112 133), often analyzed in theory
  - Shuffling (132 213), default setting in practice
  - We study shuffling since it is closer to practice

Some results claimed Adam has divergence issue

Reddi et al.18 **(ICLR Best paper):**
For any β_1, β_2 s.t. $\beta_1 < \sqrt{\beta_2}$, there exists a problem such that Adam diverges

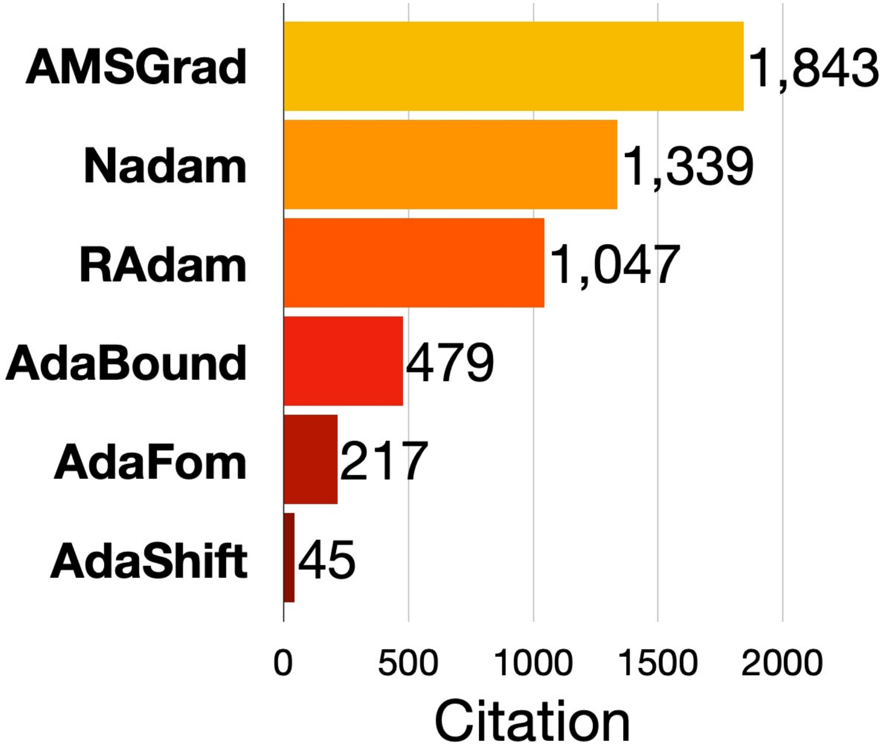


To Overcome Divergence, ⋯

- Modify Adam
  - **AMSGrad, AdaFom** [Reddi et al.'18, Chen et al.'18]: keep $v_k \geq v_{k-1}$
    - Slow convergence [Zhou et al.' 18]
  - **AdaBound** [Luo et al.' 19]: Impose constraint: $v_k \in [C_l, C_u]$
    - Need to tune two extra hyperparameters

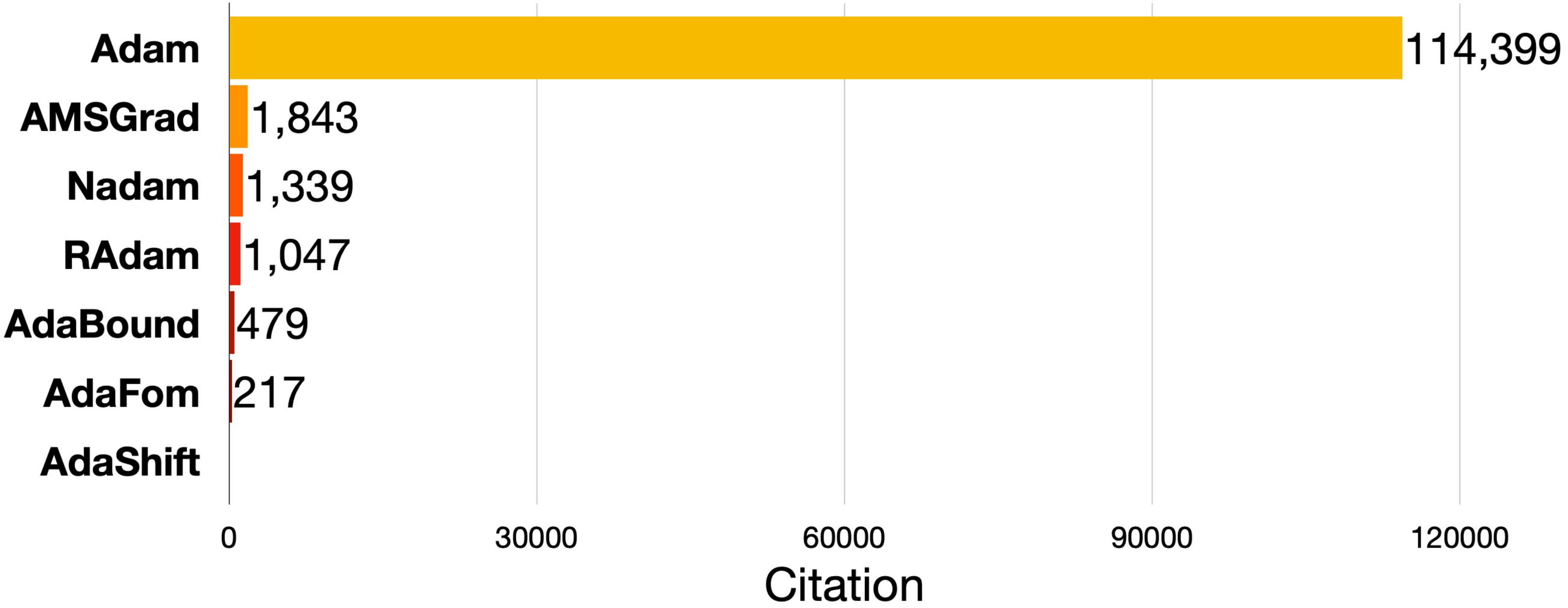
However, vanilla Adam works well for most practical applications!

Comparison: Adam vs its variants



- ∗Disclaimer: contribution is not proportional to citation.
  But citation might reflect the popularity among practitioners.

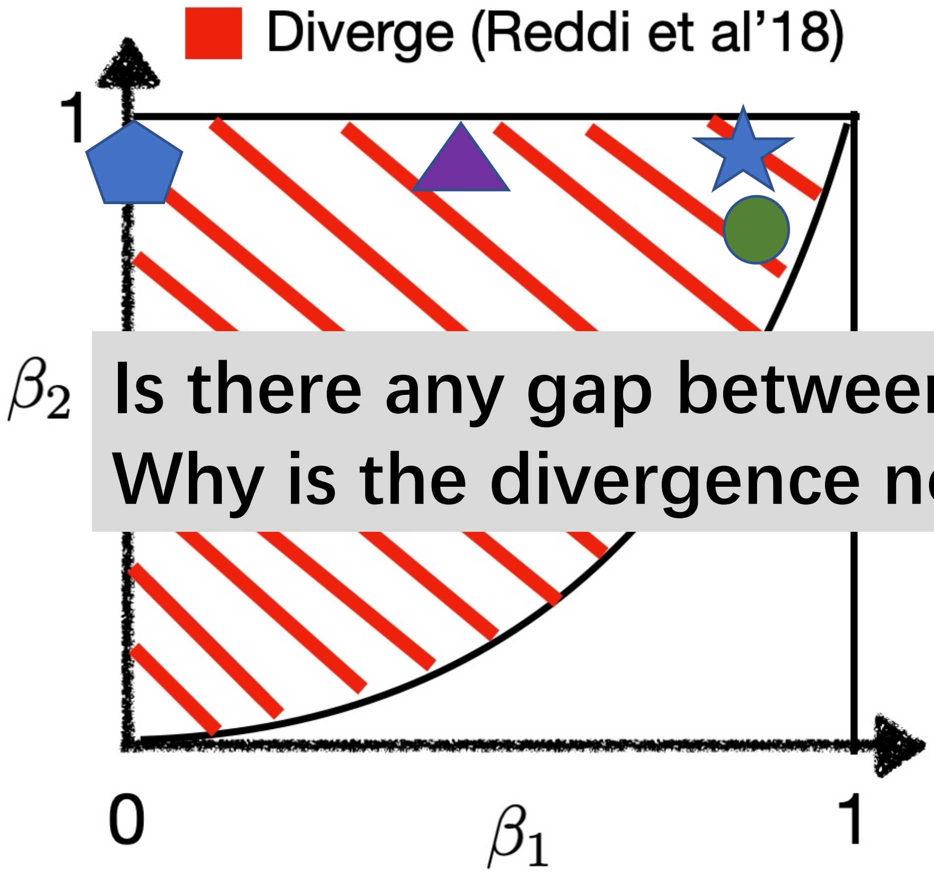
However, Adam remains overwhelmingly popular



- The attention Adam received is astonishing!
- Partially because many variants bring new issues (e.g., slow)

Divergence theory does not match practice

Observation: the reported (β_1, β_2) **actually satisfy divergence condition** $\beta_1 < \sqrt{\beta_2}$!



Is there any gap between theory and practice?
Why is the divergence not observed?

Most deep learning tasks (e.g. RL, NLP, CV, GAN, etc.):
$\beta_1 = 0.9, \beta_2 = 0.999$

l, DCGAN, etc:
.999

uage models (e.g. GPT-3)
$\beta_1 = 0.9, \beta_2 = 0.95$

First-order GAN, MSG-GAN:
$\beta_1 = 0, \beta_2 = 0.999$

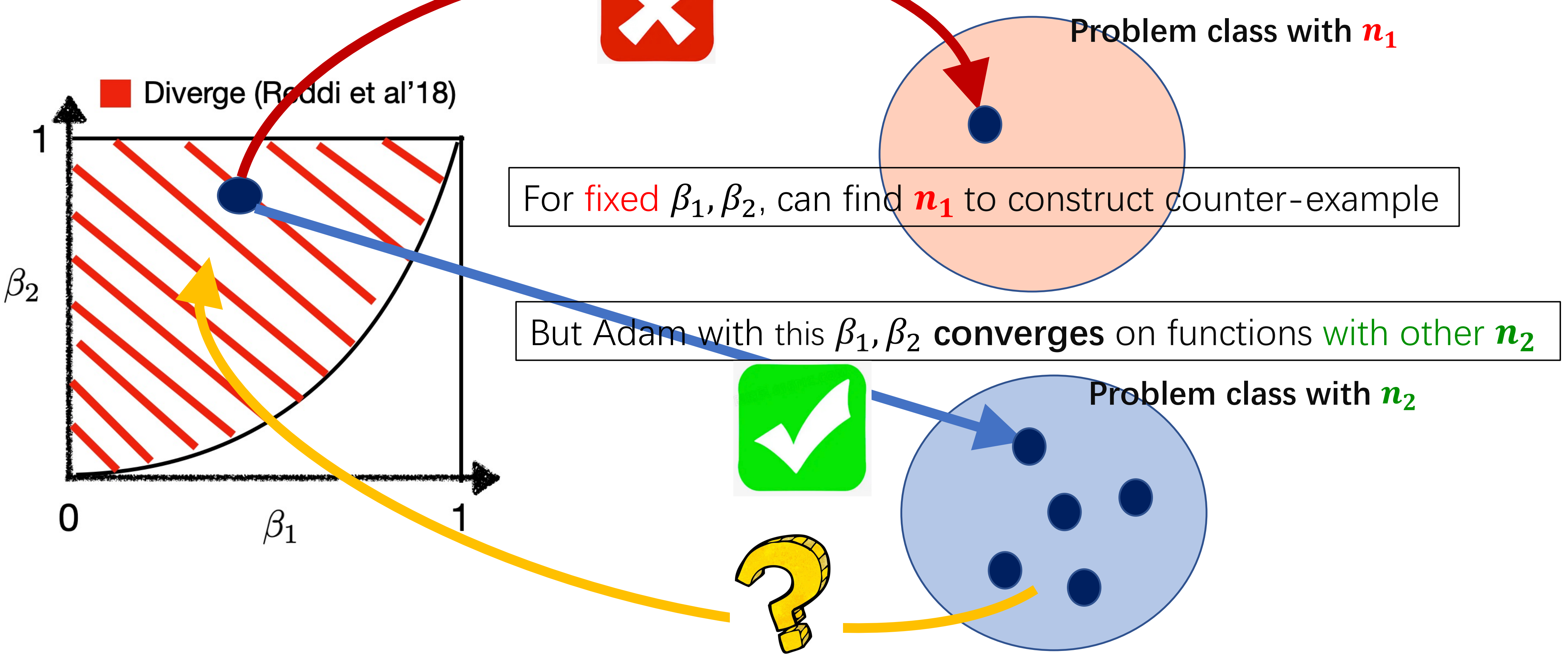
Why is divergence not observed?

- Reddi et al. 18 consider $\min_{x} f(x) := \sum_{i=1}^{n} f_i(x)$

Proof(Reddi et al. 18):
For any fixed β_1, β_2 s.t. $\beta_1 < \sqrt{\beta_2}$, we can find an n to construct the divergence example $f(x)$

- An important (but often ignored) feature: Reddi et al. fix β_1, β_2 **before picking the problem** (change n according to β_1, β_2)
- While in practice, parameters are often **problem-dependent** (e.g. tuning lr for GD)
- **Conjecture: Adam might converge for fixed problem (or fixed n)**

A simple illustration



Question: Does Adam converge for fixed problem class (fixed n)?

Contents

Assumptions

- Consider $\min_{x} \ f(x) := \sum_{i=1}^{n} f_i(x)$
- **A1 (L-smooth):** assume $\nabla f_i(x)$ are L-Lipschitz continuous.
- **A2 (Affine Variance):** $\frac{1}{n}\sum_{i=1}^{n} || \nabla f_i \ (\mathrm{x}) \ - \frac{1}{\mathrm{n}} \ \sum_{i=1}^{n} \nabla f_i(x) \ ||_2^2 \leq D_1 || \ \nabla f \ (\mathrm{x}) ||_2^2 + D_0$

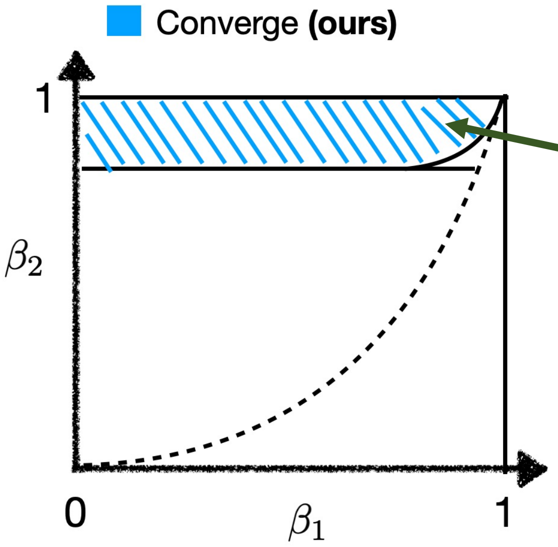
  A2 is quite general:
  - When $D_1 = 0$, A2 becomes bounded variance, commonly used in SGD analysis
    --**A2** allows $D_1 > 0$ and thus it is weaker than bounded variance.
  - When $D_0 = 0$, **A2** becomes ``Strong Growth Condition (SGC)" [Vaswani et al., 19]
    -- Intuition: When $|| \ \nabla f \ (\mathrm{x}) ||$=0 $\Longrightarrow$ we have $|| \ \nabla f_i(\mathrm{x}) ||$=0.
    -- SGC holds for overparametrized networks (Vaswani et al.19)
- We do NOT need the following assumptions, which are common in the literature
- ~~**A3** (bounded gradient):~~ $\cancel{|| \ \nabla f(x) || < C}$
- ~~**A4** (bounded 2nd order momentum):~~ $\cancel{v_k \in [C_l, C_u]}$
- To our knowledge, **A1+ A2** are the mildest assumption set so far.

Convergence results for large $\boldsymbol{\beta_2}$

- **Theorem 1:** Consider the previous setting.
  When $\beta_2 \geq 1 - O\left(\frac{1-\beta_1^n}{n^{3.5}}\right)$ and $\beta_1 < \sqrt{\beta_2} < 1$, Adam with stepsize $\eta_k = \frac{1}{\sqrt{k}}$ converges to the neighborhood of stationary points:

$$\min_{k\in[1,T]} \mathbb{E}||\nabla f(x_k)||_2^2 = O\left(\frac{\log T}{\sqrt{T}} + (1-\beta_2)D_0\right).$$

- **RK:** When $D_0 = 0$ (e.g., for overparameterized models): Adam converges to stationary points



We identify a safe region! **(UNKNOWN BEFORE!)**

Remark: Convergence to neighborhood

- When $D_0 > 0$: **converges to a neighborhood of stationary points** with size $O((1-\beta_2)D_0)$. (a.k.a. ``ambiguity zone”).
- **This** is common for

  --constant-step SGD [Yan et al., 2018; Yu et al., 2019]

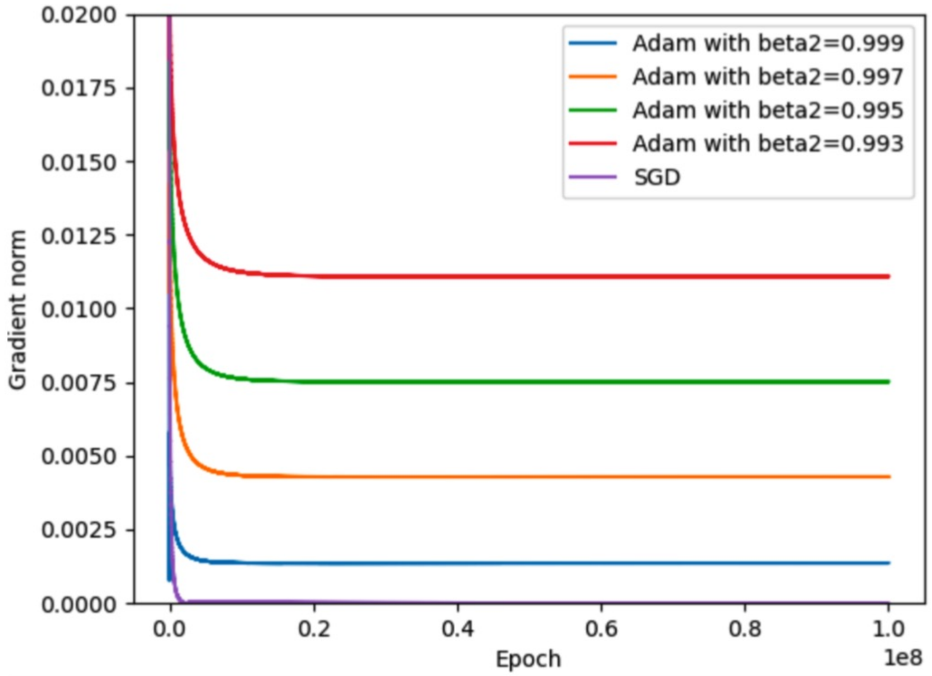
  --diminishing-lr adaptive gradient methods [Zaheer et al., 2018; Shi et al., 2020]:

$$x_{k+1} = x_k - \frac{\eta_k}{\sqrt{v_k}} m_k$$

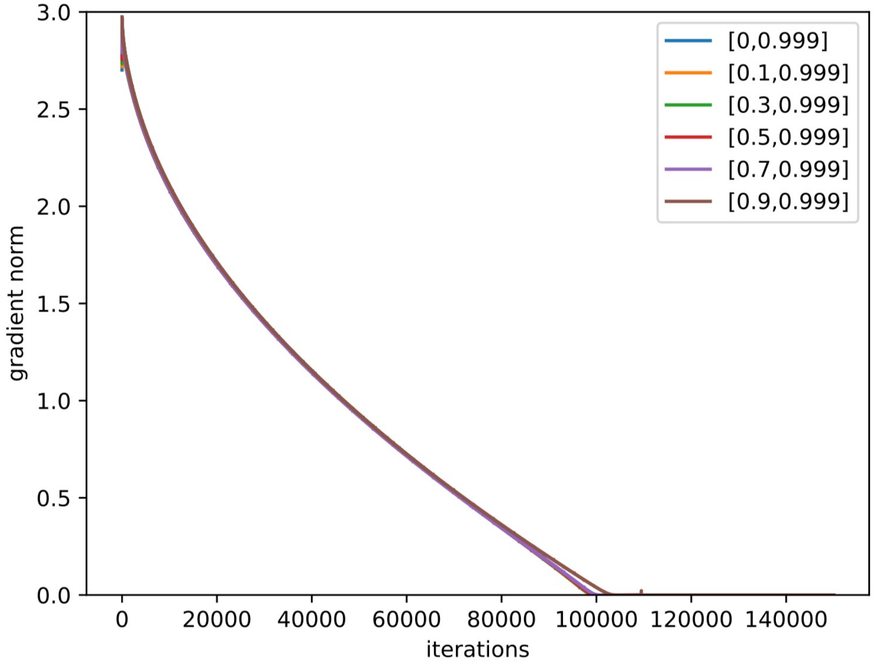
Intuition: Although η_k is diminishing, $\frac{\eta_k}{\sqrt{v_k}}$ may not decrease.

Remark: Convergence to neighborhood.

Left: An example with $D_0 > 0$



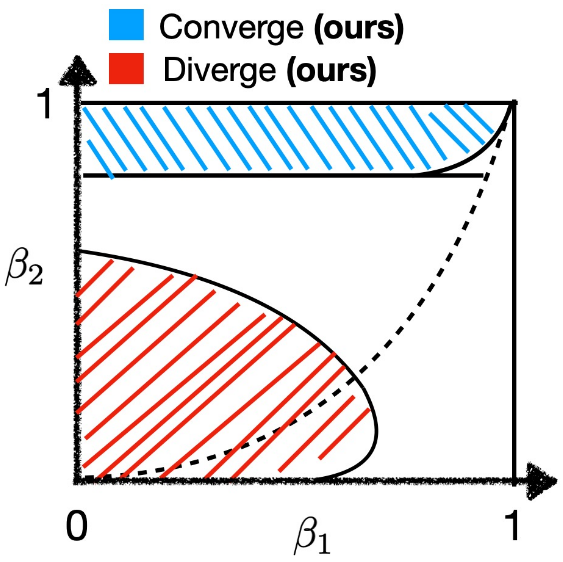
Right: An example with $D_0 = 0$



Setting: Adam & SGD with lr $\eta_k = \frac{1}{\sqrt{k}}$

How does Adam behave in the rest of the region?

- When β_2 is large: we have shown a positive result.

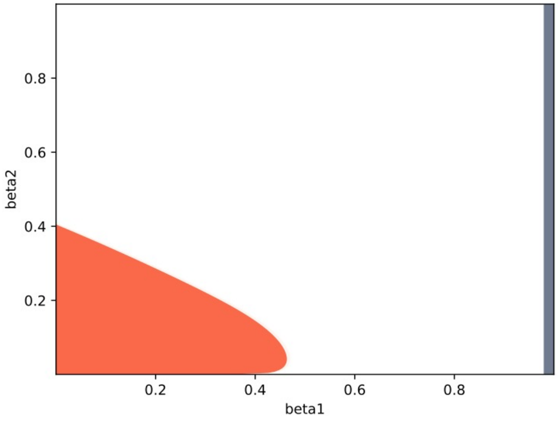


- When β_2 is small: we will show that Adam can still diverge! (even if the problem class is fixed)

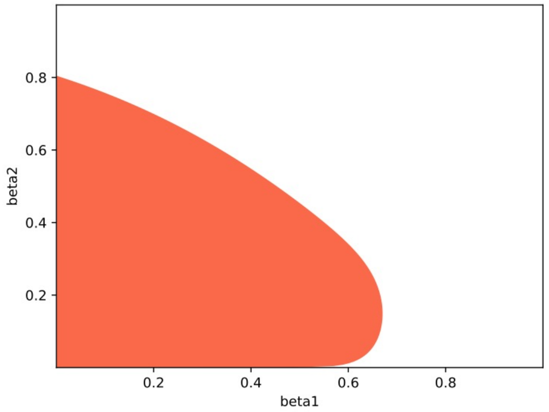
Divergence can happen when $\boldsymbol{\beta_2}$ is small

- **Thm 2:** For any fixed n, there exists a function $f(x)$ satisfying **A1** and **A2**, s.t. when (β_1, β_2) are chosen in the orange region (size depends on n), s.t. Adam' s iterates and function values diverge to infinity

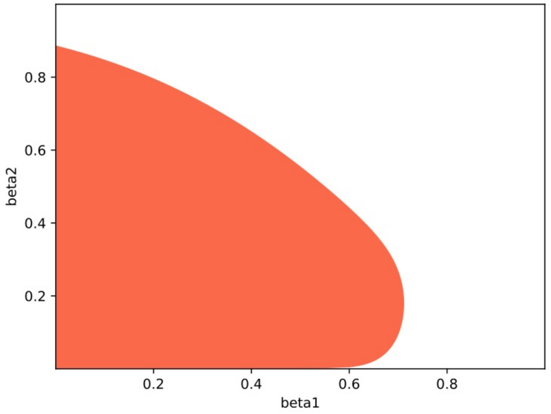
- The region is **precisely drawn** (plotted by solving some analytic conditions)
- region enlarges with n



(b) $n = 10$



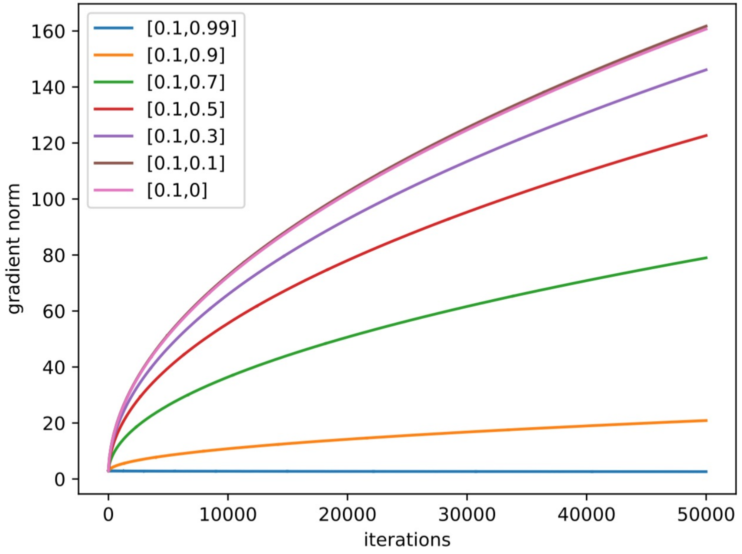
(c) $n = 50$



(d) $n = 100$

Some remarks on the divergence theorem

- **Remark 2:** For Adam, it is important to remove the bounded gradient assumption ($||\nabla f(x)|| < C$) !!!
- In practice, the gradient of iterates can be unbounded.

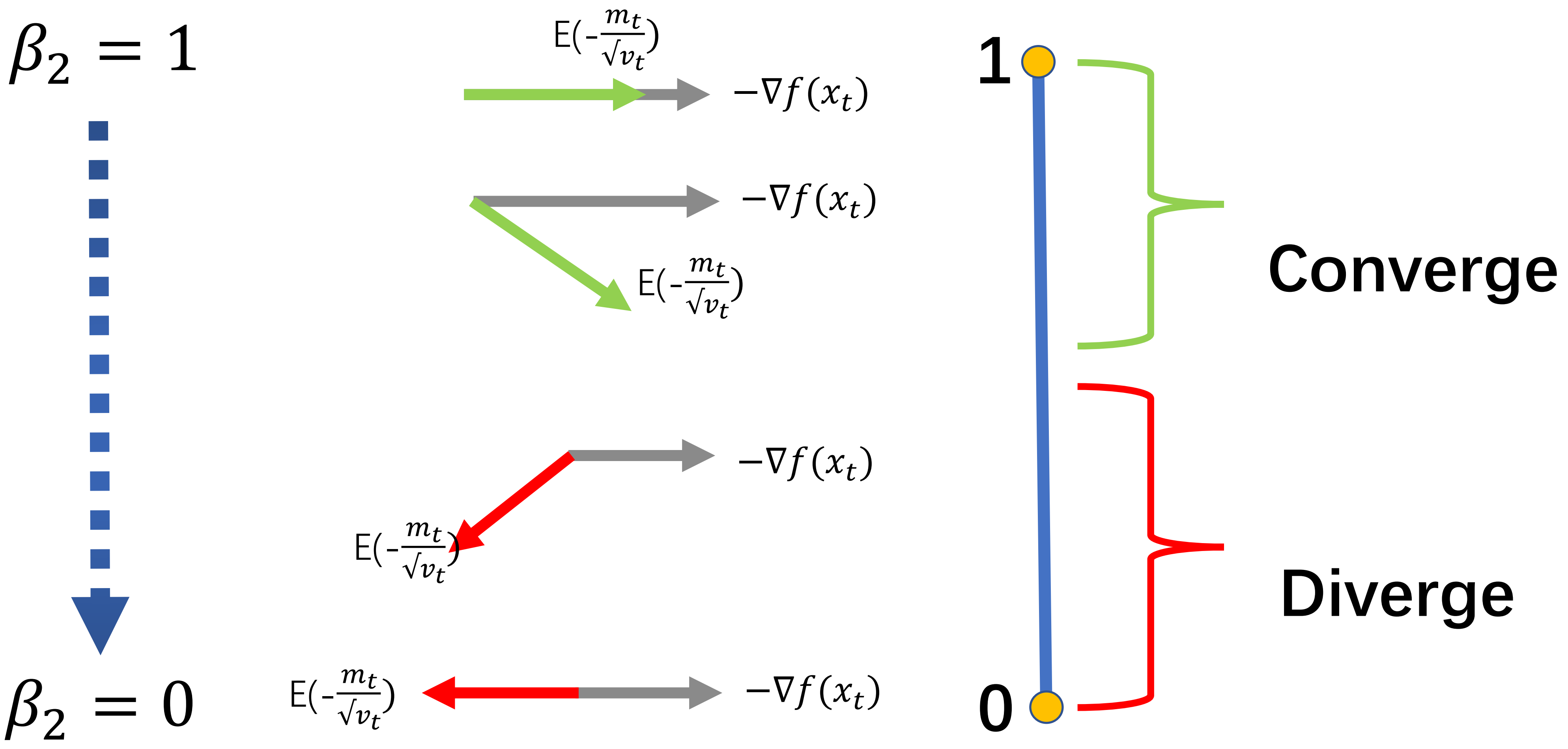


(d) $n = 20$

Contents

Intuition behind convergence and divergence

Adam: $\boldsymbol{x^{t+1} = x^{t} - \eta_t \frac{m_t}{\sqrt{v_t}}}$



Proof Ideas for Convergence Analysis: An Overview

Want to show: $\mathbb{E}\left\langle \nabla f(x_k), \frac{m_k}{\sqrt{v_k}} \right\rangle = \left\langle \nabla f(x_k), \frac{(1-\beta_1)\nabla f_{\tau_k}(x_k)+\beta_1 m_{k-1}}{\sqrt{(1-\beta_2)\nabla f_{\tau_k}(x_k)\circ\nabla f_{\tau_k}(x_k)+\beta_2 v_{k-1}}} \right\rangle > 0$

Challenge 1: m_k contains heavy history

Challenge 2: v_k brings non-linear perturbation.

Key Idea: Established a new property of Adam’s momentum under random permutations.

Step 1: Show the periodical property of momentum → Step 2: Control the perturbation when β_2 is large

Lemma 5.1. *(Informal) Consider Algorithm 1. For every $l \in [d]$ and any $\beta_1 \in [0,1)$, we have the following result under Assumption 2.1.*

$$\delta(\beta_1) := \mathbb{E}\sum_{i=0}^{n-1}\left(m_{l,k,i} - \partial_l f_{\tau_{k,i}}(x_{k,0})\right) = \mathcal{O}\left(\frac{1}{\sqrt{k}}\right),$$

where $\partial_l f(x_{k,0})$ is the l-th component of $\nabla f(x_{k,0})$; $m_{l,k,i} = (1-\beta_1)\partial_l f_{\tau_{k,i}}(x_{k,i}) + \beta_1 m_{l,k,i-1}$.

Lemma 5.2. *(Informal) Under Assumption 2.1 and 2.2, consider Algorithm 1 with large β_2 and $\beta_1 < \sqrt{\beta_2}$. For those l with gradient component larger than certain threshold, we have:*

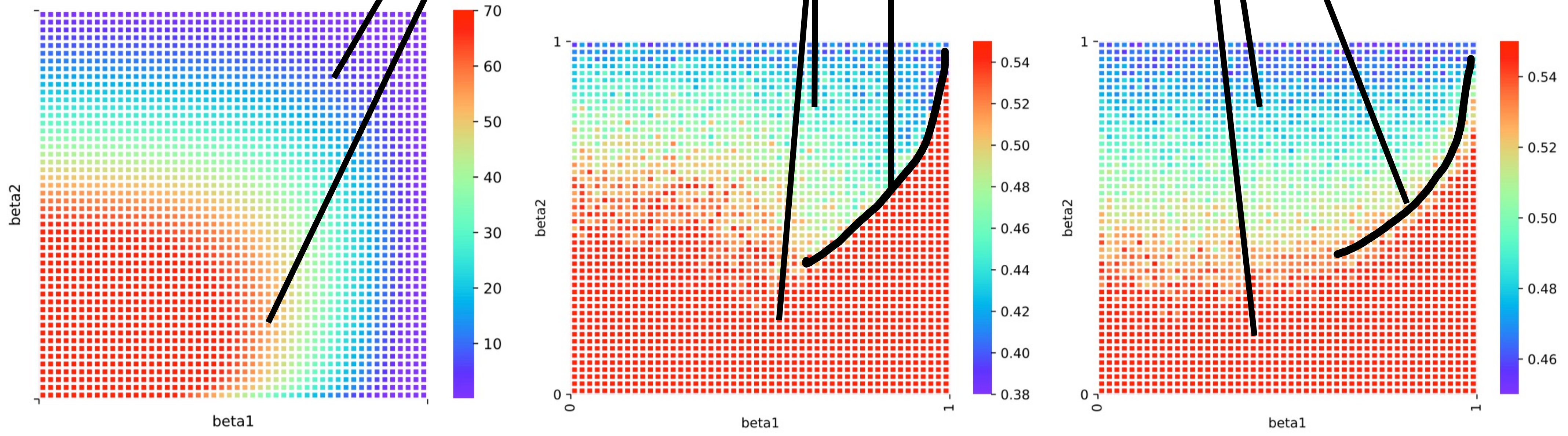
$$\left|\frac{\partial_l f(x_{k,0})}{\sqrt{v_{k,0}}} - \frac{\partial_l f(x_{k-1,0})}{\sqrt{v_{k-1,0}}}\right| = \mathcal{O}\left(\frac{1}{\sqrt{k}}\right); \tag{5}$$

$$\mathbb{E}\left(\frac{\partial_l f(x_{k,0})}{\sqrt{v_{l,k,0}}}\sum_{i=0}^{n-1}\left(m_{l,k,i} - \partial_l f_{\tau_{k,i}}(x_{k,0})\right)\right) = \mathcal{O}\left(\frac{1}{\sqrt{k}}\right). \tag{6}$$

Contents

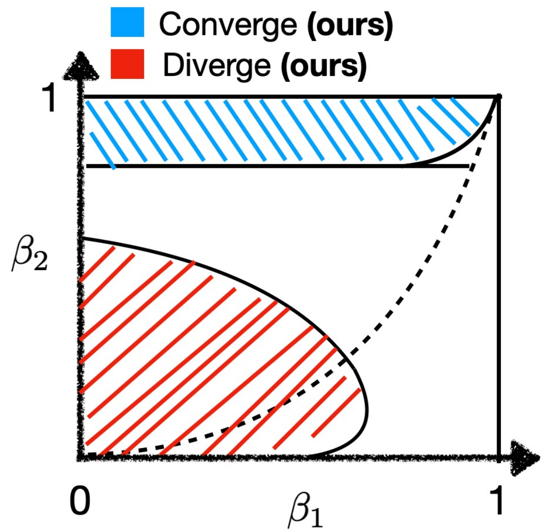
Our theory is consistent with experiments

Optim Optimization error is Smooth boundaries ion



(a) Function (2) (b) MNIST (c) CIFAR-10

Summary: the behavior of Adam changes dramatically under different hyperparameters



When increasing β_2:
There is a phase transition from divergence to convergence.

Setting	Hyperparameters	Adam’s behavior
$\forall f(x)$ under **A1** and **A2** with $D_0 = 0$	β_2 is large and $\beta_1 < \sqrt{\beta_2}$	Converges to stationary points **(Ours)**
$\forall f(x)$ under **A1** and **A2** with $D_0 > 0$	β_2 is large and $\beta_1 < \sqrt{\beta_2}$	Converges to the neighborhood of stationary points **(Ours)**
$\exists\ f(x)$ under **A1** and **A2**	β_2 is small and a wide range of β_1	Diverges to infinity **(Ours)**

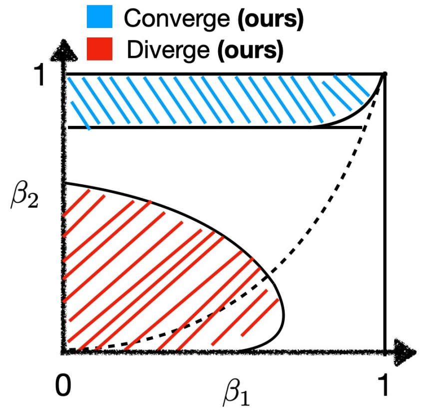
Implication to practitioners

- **Case study:** Bob is using Adam to train NNs. However, Adam with default hyperparameter fails in his tasks.
- Bob heard there is a well-known result that Adam can diverge.
- So he wonders: shall I keep tuning hyperparameter to make it work?
- Or shall I just give up and switch to other algorithms like AdaBound (which has 2 extra hyperparameters)?

Our suggestions:

1. Adam is still a theoretically justified algorithm. **Please use it confidently!**
2. Suggestions for hyperparameter tunning:
   In one sentence: First, tune up β_2. Then, try different β_1 with $\beta_1 < \sqrt{\beta_2}$
   In details: see next page

Receipe for hyperparameter-tuning



AdaShift? AdaBound?

No need!

Change algorithm?

Adam with default setting $(\beta_1, \beta_2) = (0.9, 0.999)$ **fails** in your task

Tune up β_2.
Try 0.9995, 0.9999, 0.99995, etc

Does not work better?

Try smaller β_1.
Try 0.7, 0.5, 0.3, 0.1, etc

Does not work better?

Adam with default setting $(\beta_1, \beta_2) = (0.9, 0.999)$ **works** in your task, but want better performance

Tune down β_2 a bit.
Try 0.995, 0.99, 0.95, etc
But not too small!

Mainly based on:

[1] Adam Can Converge Without Any Modification on Update Rules
(NeurIPS 2022)

Yushun Zhang, Congliang Chen, Naichen Shi, Ruoyu Sun, Zhi-Quan Luo

Thanks to all the collaborators!