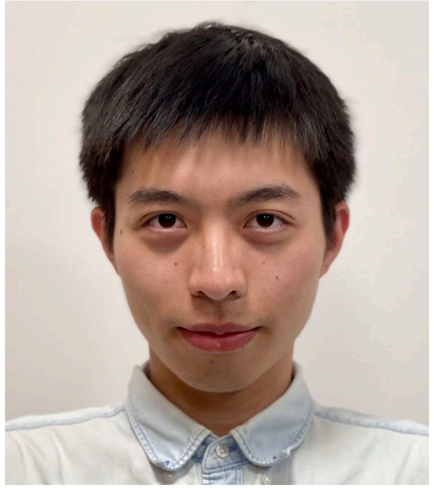
Adam Can Converge Without Any Modification On Update Rules

Yushun Zhang[1], Congliang Chen[1], Naichen Shi[2], Ruoyu Sun[1], Zhi-Quan Luo[1]

1: The Chinese University of Hong Kong, Shenzhen, China; 2: University of Michigan, US

Motivation

- **Adam** is one of the most popular algorithms in deep learning (DL).
  (It has received more than 110,000 citations)

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

Preliminaries on Adam

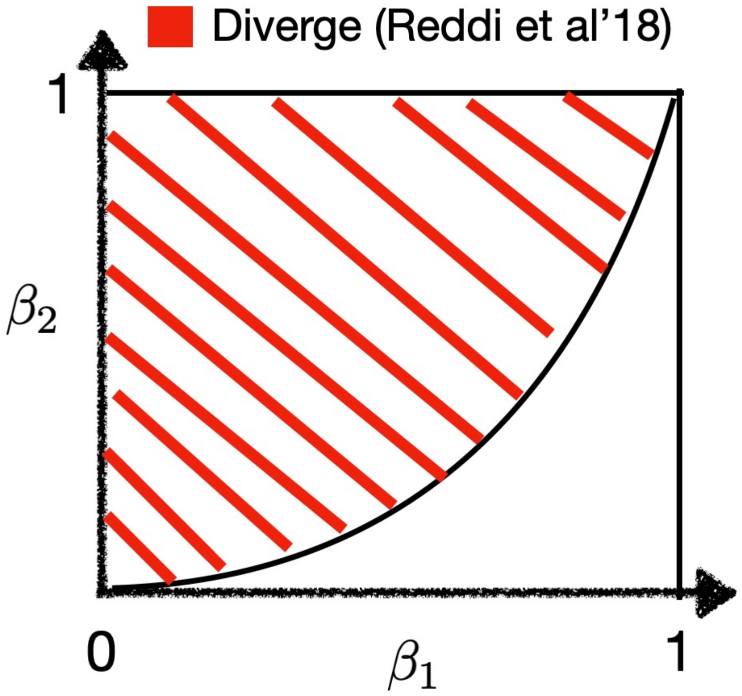
- Consider unconstrained problem $\min_x f(x) := \sum_{i=1}^n f_i(x)$
- In deep learning (DL) tasks, n often stands for sample size; x denotes trainable parameters
- Initialization $\nabla f(x_0)$, $m_0 = \nabla f(x_0)$
- In the k-th iteration: sample τ_k from $\{1,2,\dots,n\}$

- Adaptive Momentum Estimator (Adam) [Kingma and Ba 15]:
- $m_k = (1-\beta_1)\nabla \mathrm{f}_{\tau_k}(\mathrm{x}_k) + \beta_1 m_{k-1}$
- $v_k = (1-\beta_2)\nabla \mathrm{f}_{\tau_k}(\mathrm{x}_k) \circ \nabla \mathrm{f}_{\tau_k}(x_k) + \beta_2 v_{k-1}$
- $x_{k+1} = x_k - \eta_k \frac{m_k}{\sqrt{v_k}}$

- Notations: $\circ$, $\sqrt{}$, and division are all element-wise operations.
- 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: $\beta_2 = 0.999$

For A Long Time, Adam Is Criticized For Its 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

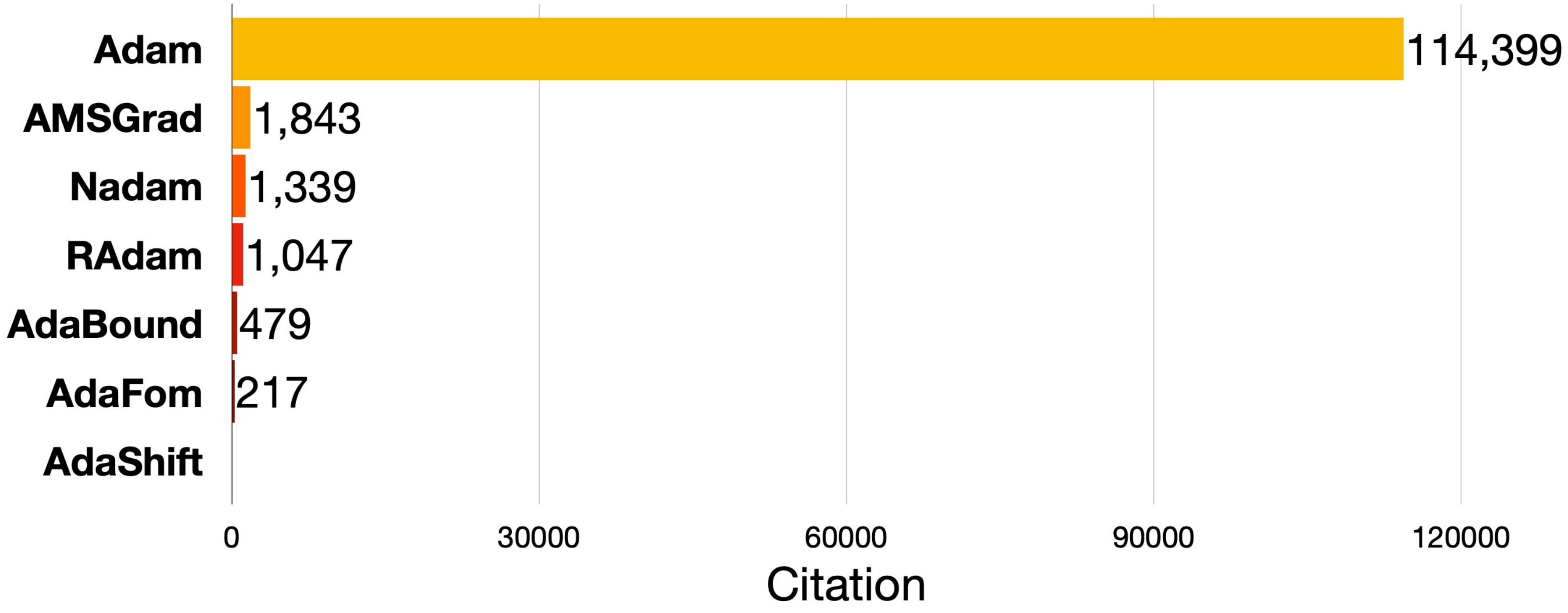


How to Ensure Convergence?

- A popular line of work: Modify the algorithm! For instance:
  - **AMSGrad, AdaFom** [Reddi et al. 18, Chen et al.18]: keep $v_k \geq v_{k-1}$
  - **AdaBound** [Luo et al. 19]: Impose constraint: $v_k \in [C_l, C_u]$

- Although these Adam-variants fix the divergence issue, they often bring new issues. For instance:
  - **AMSGrad and AdaFom** are reported to be slow [Zhou et al. 18].
  - **AdaBound** introduces 2 extra hyperparameters.
- On the other hand, **Adam remains exceptionally popular. It works well in practice!** (either under default setting, or after proper tuning).

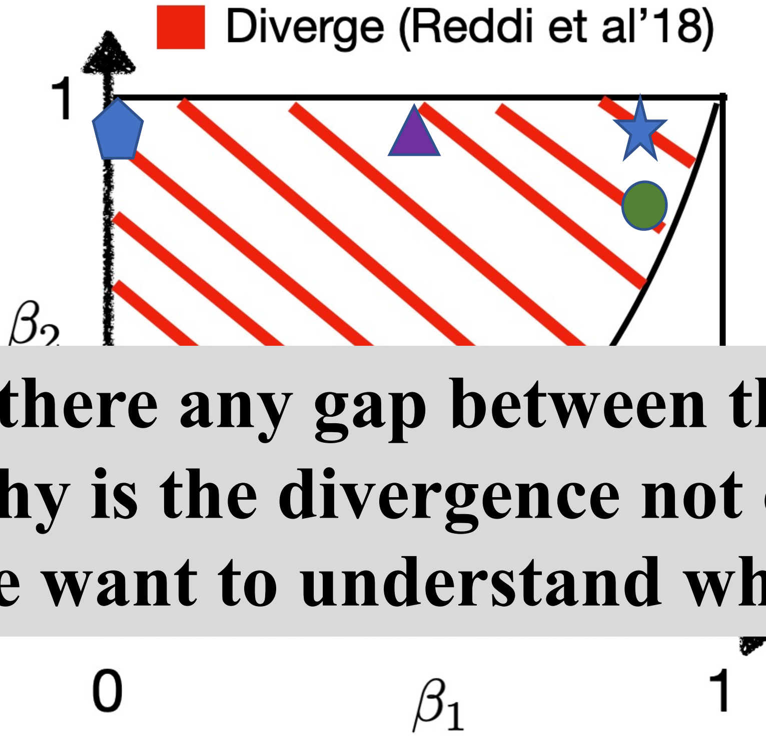
Adam Remains Exceptionally Popular Among Practitioners



- Though being criticized for divergence issue, Adam is still becoming one of the most influential algorithms in deep learning
- Partially because the new variants often bring new issues (e.g. converge slowly)
- *Disclaimer: contribution is not necessarily proportional to citations.

Further, The Divergence Theory Does Not Go Well with Practice

We find that the reported (β_1, β_2) in the successful applications **actually satisfy the divergence condition** $\beta_1 < \sqrt{\beta_2}$!



- Most deep learning tasks (e.g. RL, NLP, CV, GAN, etc.): $\beta_1 = 0.9, \beta_2 = 0.999$
- Conditional GAN, DCGAN, etc: $_2 = 0.999$
- language models (e.g. GPT-3) $= 0.95$
- First-order GAN, MSG-GAN: $\beta_1 = 0, \beta_2 = 0.999$

Is there any gap between theory and practice?
Why is the divergence not observed?
We want to understand why.

We Revisit The Counter-example In Reddi et al. 18

- 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 (n is changing).
- While in optimization field, parameters are often **problem-dependent** (e.g. the step size for GD). As such, the divergence is hardly surprising.

Conjecture: Adam might converge under fixed problem (fixed n).

Our Results: Adam Can Converge Without Any Modification

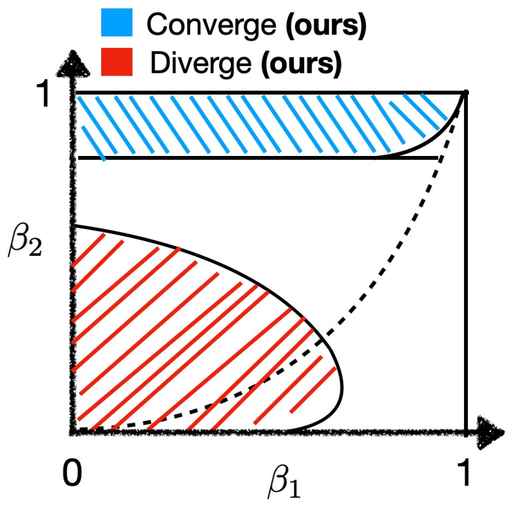
Theorem 1: Given problem $\min_{x} f(x) := \sum_{i=1}^{n} f_i(x)$, under the practical assumptions (mildest so far), we prove that：when $\beta_2 \geq 1 - O\left(\frac{1-\beta_1^n}{n^{3.5}}\right)$, $\beta_1 < \sqrt{\beta_2} < 1$, **Adam converges with rate $O(\frac{\log k}{\sqrt{k}})$.***

Theorem 2: Consider the same setting as above, $\exists f(x)$, s.t., when (β_1, β_2) lies in the red region, the sequence $\{x_k\}$ and $\{f(x_k)\}$ of Adam diverges to ∞.

Remark 1: Our convergence results covers any $\beta_1 \in [0,1)$, including the default setting. This is the first result showing that Adam can converge without any modification on its update rules.

Remark 2: We do not need stronger assumptions like bounded gradient assumptions ($||\nabla f(x)|| < C$) or bounded adaptor ($v_k \in [C_l, C_u]$).

Proof Idea: Identify certain periodic property of Adam’s momentum under random permutations and non-linear dynamics.



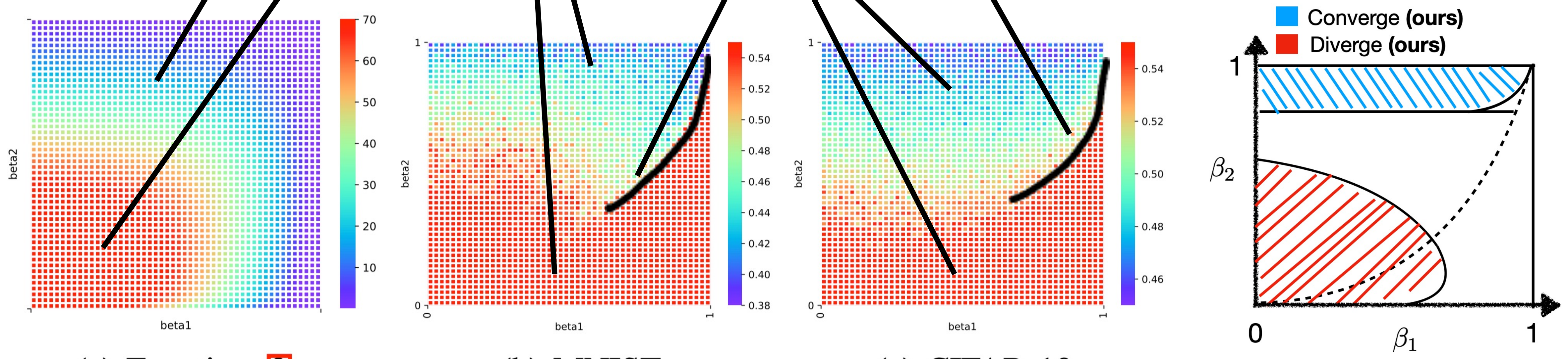
*: We further distinguish two sub-cases: convergence to neighborhood of stationary points and to the exact stationary points. Please check our paper for more detailed characterization.

Our Theory Matches Experiments

Small optimization error in the b

Smooth boundaries

Large optimization error in the red region



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

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:**
   First, tune up β_2. Then, try different β_1 with $\beta_1 < \sqrt{\beta_2}$