

Learning with SGD and Random Features

Luigi Carratino - University of Genoa

Alessandro Rudi - INRIA / ENS

Lorenzo Rosasco - University of Genoa, IIT, MIT

Supervised learning

Given $(x_1, y_1), \dots, (x_n, y_n)$

Learn a **non-linear** function $f : X \rightarrow Y$ e.g. $f(x) = w^\top \phi(x)$

Goal: f provably **accurate** + computationally **efficient**

Learning with Stochastic Gradients & Random Features



$$w_{t+1} = w_t - \gamma \nabla L (w_t^\top \phi_M(x_t), y_t)$$

loss function



$$\phi_M(x) \rightarrow \begin{bmatrix} \sigma(s_1^\top x) \\ \vdots \\ \sigma(s_M^\top x) \end{bmatrix}$$

non-linear function

random sketching

SGD-RF with Mini Batching

$$w_{t+1} = w_t - \underbrace{\gamma}_{\text{stepsize}} \frac{1}{\underbrace{b}_{\text{batchsize}}} \sum_{i=\underbrace{b(t-1)+1}^{bt}} \nabla L(w_t^\top \phi_{\underbrace{M}_{\text{n}^\circ \text{ random features}}}(x_{i_t}), y_{i_t}) \quad t = 1, \dots, \underbrace{T}_{\text{n}^\circ \text{ iterations}}$$

Complexity: Time $\mathcal{O}(MbT)$ Space $\mathcal{O}(M)$

How to choose γ, b, M, T for optimal accuracy?

Our main result

Theorem(Carratino, Rudi, Rosasco 2018)

Let L be the squared loss

$$\underbrace{\mathbb{E}_{x,y} L(w_t^\top \phi_M(x), y) - \inf_w \mathbb{E}_{x,y} L(w^\top \phi_\infty(x), y)}_{\text{“Test error”}} \lesssim \frac{\gamma}{b} + \left(\frac{\gamma t}{M} + 1 \right) \frac{\gamma t}{n} + \frac{1}{\gamma t} + \frac{1}{M}$$

“Test error”

Optimize w.r.t γ, b, M, T to get the best rate

Recipe

$$\text{“Test error”} \lesssim \frac{1}{\sqrt{n}}$$

Take $M = \sqrt{n}$

- $b = 1$ SGD single pass

$$\implies \gamma = \frac{1}{\sqrt{n}}$$

- $b = \sqrt{n}$ mini-batch single pass

$$\implies \gamma = 1$$

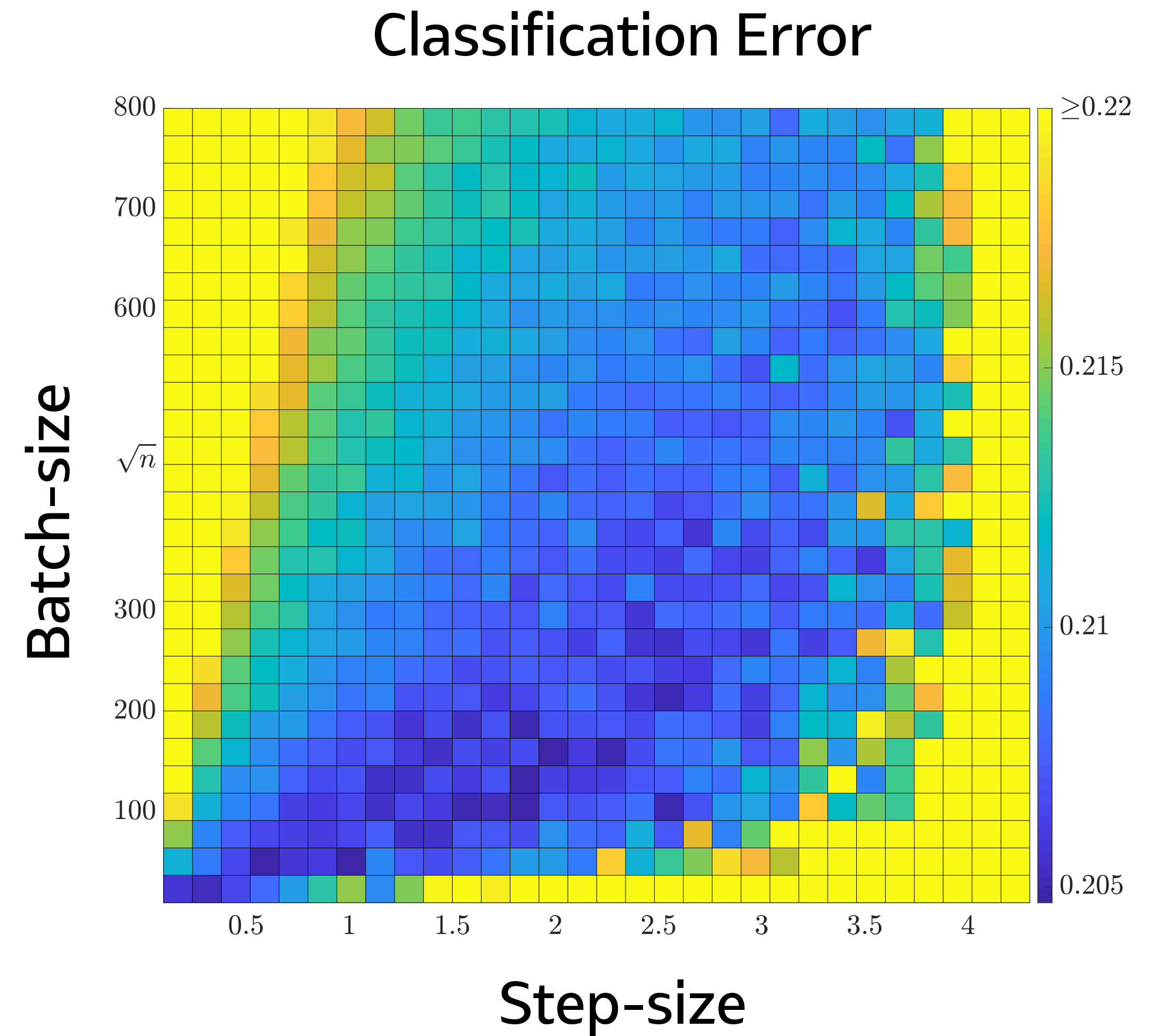
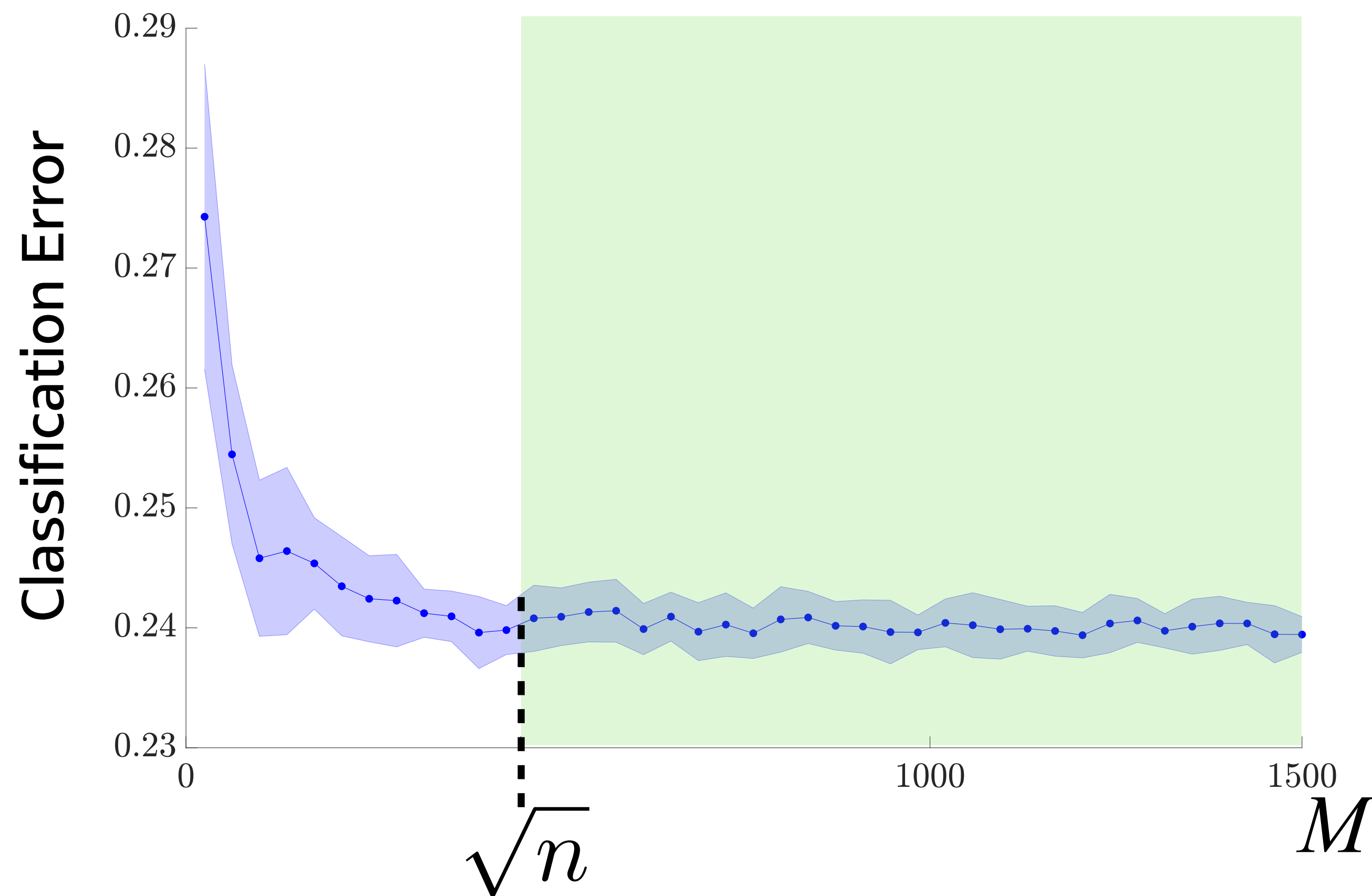
Complexity:

Time $\mathcal{O}(n\sqrt{n})$

Space $\mathcal{O}(\sqrt{n})$

Practice validates theory

SUSY dataset $n \approx 10^6$



Contribution

SGD-RF leads to optimal accuracy
with minimum computation

+ Faster rates

Poster #127

+ Decreasing step-size