Breakout 3: RandNLA for AI and Data-Intensive Science

University of Wisconsin, Madison
May 12, 2026

Session Participants: Stephen Becker, Petros Drineas, Keith Levin, Qin Li, Michael Mahoney, Kevin Miller, Garvesh Raskutti, Shambhavi Suryanarayanan, Yifu Wang

Overview

How can RandNLA be transitioned from a theoretical tool to a fundamental component of the AI/ML pipeline?

We summarized:

- existing techniques in AI/ML software
- practical hurdles to widespread adoption
- how RNLA can become a "first-class citizen" in AI/ML

Existing Methods

Several algorithms are already available in the APIs of major software frameworks for data science and AI/ML

Examples:

- Python/PyTorch
  - `scikit-learn`: includes RSVD and data-oblivious randomized subspace embeddings within the module
  - PyTorch-native extensions `torch_linops` and BACKPACK also include RandNLA primitives (e.g. randomized trace estimation)
  - Panther: attention mechanism with RSVD, randomized sketching
- Julia
  - Flux.jl: relies on `RandomizedLinAlg.jl` for randomized low-rank approximation and matrix factorizations

Preliminaries

We should begin by asking "the right questions":

Preliminaries

We should begin by asking "the right questions":

- What is the problem we are trying to solve? Does it require domain expertise in another field?

Preliminaries

We should begin by asking "the right questions":

- What is the problem we are trying to solve? Does it require domain expertise in another field?
- How can we make computational pipelines more problem-agnostic?

Preliminaries

We should begin by asking "the right questions":

- What is the problem we are trying to solve? Does it require domain expertise in another field?
- How can we make computational pipelines more problem-agnostic?
- What are the most crucial or high-impact insertion points in AI/ML pipelines?

Preliminaries

We should begin by asking "the right questions":

- What is the problem we are trying to solve? Does it require domain expertise in another field?
- How can we make computational pipelines more problem-agnostic?
- What are the most crucial or high-impact insertion points in AI/ML pipelines?
- How do we train RandNLA researchers on subjects outside of math/CS and reward people for making that leap?

Further development of RandNLA for AI/ML

We discussed the following ways to help fully integrate RandNLA into AI/ML pipelines:

- More communication with domain experts and practitioners of AI/ML

Further development of RandNLA for AI/ML

We discussed the following ways to help fully integrate RandNLA into AI/ML pipelines:

- More communication with domain experts and practitioners of AI/ML
- More funding and opportunities for interdisciplinary work/grants/centers

Further development of RandNLA for AI/ML

We discussed the following ways to help fully integrate RandNLA into AI/ML pipelines:

- More communication with domain experts and practitioners of AI/ML
- More funding and opportunities for interdisciplinary work/grants/centers
- More incentives on the university side for math/CS/stats researchers to acquire domain knowledge for SciML problems

Further development of RandNLA for AI/ML

We discussed the following ways to help fully integrate RandNLA into AI/ML pipelines:

- More communication with domain experts and practitioners of AI/ML
- More funding and opportunities for interdisciplinary work/grants/centers
- More incentives on the university side for math/CS/stats researchers to acquire domain knowledge for SciML problems
- More research on burgeoning areas of RandNLA (e.g. quantum computing)