My Experiences with AI in Mathematics

Nupur Jain

Ruhr-Universität Bochum

Benchmarks in Leipzig, 2026

nupurj.github.io/experiences-ai

Goal of this talk

What this talk is not:

- How to use LLMs or Lean to prove theorems
- A technical introduction to machine learning

Goal of this talk

What this talk is not:

- How to use LLMs or Lean to prove theorems
- A technical introduction to machine learning

What this talk is about:

- What kind of machine learning techniques can I use in my math research?
- How can machine learning be useful to me? What kind of problems can I attack with machine learning?
- What might a typical workflow using machine learning in math research look like?
- How can I get started?

Why use machine learning?

Perhaps you use programming languages (Python, SAGE, Oscar...) and databases (OEIS, FindStat...) for:

- sanity checks
- computing large examples you can't do by hand
- testing out a conjecture
- generating data to look for patterns
- checking if something you are looking for already exists
- ...

Why use machine learning?

Perhaps you use programming languages (Python, SAGE, Oscar...) and databases (OEIS, FindStat...) for:

- sanity checks
- computing large examples you can't do by hand
- testing out a conjecture
- generating data to look for patterns
- checking if something you are looking for already exists
- ...

Machine learning can be another tool in your box for:

- Search for an example that occurs sparsely over large spaces
- Learn hidden patterns in mathematical structures
- Symbolic formula discovery
- Pushing the bounds on an inequality
- ...

Who am I?

Article

Advancing mathematics by guiding human intuition with AI

https://doi.org/10.1038/s41586-021-04086-x

Received: 10 July 2021

Accepted: 30 September 2021

Published online: 1 December 2021

Open access

Check for updates

Alex Davies[1✉], Petar Veličković[1], Lars Buesing[1], Sam Blackwell[1], Daniel Zheng[1], Nenad Tomašev[1], Richard Tanburn[1], Peter Battaglia[1], Charles Blundell[1], András Juhász[2], Marc Lackenby[2], Geordie Williamson[3], Demis Hassabis[1] & Pushmeet Kohli[1✉]

The practice of mathematics involves discovering patterns and using these to formulate and prove conjectures, resulting in theorems. Since the 1960s, mathematicians have used computers to assist in the discovery of patterns and formulation of conjectures[1], most famously in the Birch and Swinnerton-Dyer conjecture[2], a Millennium Prize Problem[3]. Here we provide examples of new fundamental results in pure mathematics that have been discovered with the assistance of machine learning—demonstrating a method by which machine learning can aid mathematicians in discovering new conjectures and theorems. We propose a process of using machine learning to discover potential patterns and relations between mathematical objects, understanding them with attribution techniques and using these observations to guide intuition and propose conjectures. We outline this machine-learning-guided framework and demonstrate its successful application to current research questions in distinct areas of pure mathematics, in each case showing how it led to meaningful mathematical contributions on important open problems: a new connection between the algebraic and geometric structure of knots, and a candidate algorithm predicted by the combinatorial invariance conjecture for symmetric groups[4]. Our work may serve as a model for collaboration between the fields of mathematics and artificial intelligence (AI) that can achieve surprising results by leveraging the respective strengths of mathematicians and machine learning.

One of the central drivers of mathematical progress is the discovery of patterns and formulation of useful conjectures: statements that are suspected to be true but have not been proven to hold in all cases. Mathematicians have always used data to help in this process—from

that AI can also be used to assist in the discovery of theorems and conjectures at the forefront of mathematical research. This extends work using supervised learning to find patterns[20–24] by focusing on enabling mathematicians to understand the learned functions and derive useful

MATHEMATICAL EXPLORATION AND DISCOVERY AT SCALE

BOGDAN GEORGIEV, JAVIER GÓMEZ-SERRANO, TERENCE TAO, AND ADAM ZSOLT WAGNER

ABSTRACT. `AlphaEvolve`, introduced in [224], is a generic evolutionary coding agent that combines the generative capabilities of LLMs with automated evaluation in an iterative evolutionary framework that proposes, tests, and refines algorithmic solutions to challenging scientific and practical problems. In this paper we showcase `AlphaEvolve` as a tool for autonomously discovering novel mathematical constructions and advancing our understanding of long-standing open problems.

To demonstrate its breadth, we considered a list of 67 problems spanning mathematical analysis, combinatorics, geometry, and number theory. The system rediscovered the best known solutions in most of the cases and discovered improved solutions in several. In some instances, `AlphaEvolve` is also able to *generalize* results for a finite number of input values into a formula valid for all input values. Furthermore, we are able to combine this methodology with `Deep Think` [149] and `AlphaProof` [148] in a broader framework where the additional proof-assistants and reasoning systems provide automated proof generation and further mathematical insights.

These results demonstrate that large language model-guided evolutionary search can autonomously discover mathematical constructions that complement human intuition, at times matching or even improving the best known results, highlighting the potential for significant new ways of interaction between mathematicians and AI systems. We present `AlphaEvolve` as a powerful tool for mathematical discovery, capable of exploring vast search spaces to solve complex optimization problems at scale, often with significantly reduced requirements on preparation and computation time.

1. INTRODUCTION

The landscape of mathematical discovery has been fundamentally transformed by the emergence of computational tools that can autonomously explore mathematical spaces and generate novel constructions [56, 120, 242, 291]. `AlphaEvolve` (see [224]) represents a step in this evolution, demonstrating that large language models, when combined with evolutionary computation and rigorous automated evaluation, can discover explicit constructions that either match or improve upon the best-known bounds to long-standing mathematical problems, at large scales.

- Kissing numbers
- Sphere packing
- Sidorenko's conjecture
- The prime number theorem
- Erdős discrepancy problem
- Moving sofa problem
- Erdős–Szekeres Happy Ending problem
- Triangle density in graphs
- Heilbronn problems
- Kakeya needle problem
- ...

Resources

- https://seewoo5.github.io/awesome-ai-for-math/

All Subjects | All Years | Search title, subject, or venue... | Clear | Showing 151 of 151 papers

Title ▲	Subject(s)	Venue ▲	Year ▼	Links
ABC implies that Ramanujan's tau function misses almost all primes	Number Theory, ATP	arXiv	2026	Code
Agentic Neurosymbolic Collaboration for Mathematical Discovery: A Case Study in Combinatorial Design	Combinatorics, LLM, ATP	arXiv	2026	
AI Co-Mathematician: Accelerating Mathematicians with Agentic AI	Group Theory, Representation Theory, LLM	arXiv	2026	
Aletheia tackles FirstProof autonomously	LLM, Benchmark	arXiv	2026	Chat Logs
Almost all primes are partially regular	Number Theory, ATP	arXiv	2026	Code
Arithmetic volumes of moduli stacks of Shtukas	Number Theory, Algebraic Geometry, LLM	arXiv	2026	Chat Logs
Artificial Intelligence and the Structure of Mathematics	Survey	arXiv	2026	
Automated Conjecture Resolution with Formal Verification	LLM, ATP	arXiv	2026	Code (Rethlas) Code (Archon) Code (Lean)
CayleyPy RL: Pathfinding and Reinforcement Learning on Cayley Graphs	Graph Theory, Group Theory, RL	Advances in Theoretical and Mathematical Physics	2026	Code arXiv
Counting partial Hadamard matrices in the cubic regime	Combinatorics, LLM	arXiv	2026	
Dead ends in square-free digit walks	Number Theory, ATP	arXiv	2026	Code
Deep Reinforcement Learning for Fano Hypersurfaces	Algebraic Geometry, RL	arXiv	2026	Code
Doubly Saturated Ramsey Graphs: A Case Study in Computer-Assisted Mathematical Discovery	Combinatorics, LLM, ATP	arXiv	2026	

Mode 1: Prediction

The question

Can a neural network learn new relationships between combinatorial statistics?

- Permutation statistics on FindStat (~ 400)
- Idea: train a model to predict one statistic from the others
  - if it could, then perhaps we could uncover a previously-unknown relationship between these statistics
- Tried: neural networks, random forests, transformers...

Issues

Combinatorial statistics are highly discontinuous!

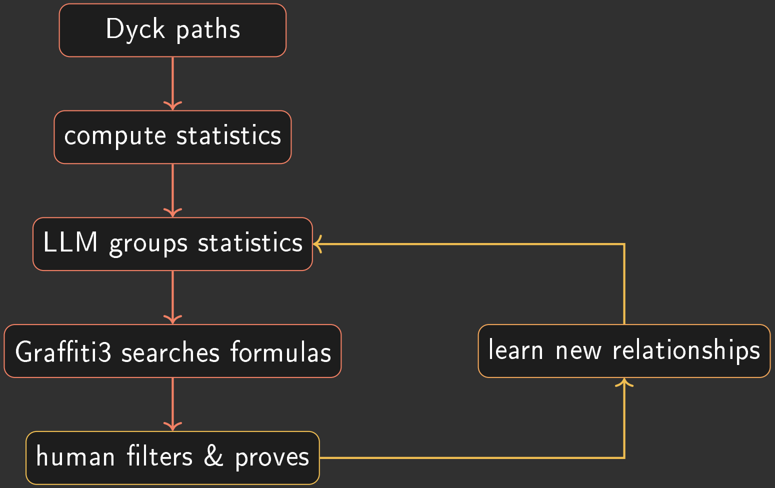
Mode 2: Automated Conjecturing

The question

Can we find **formulas** connecting combinatorial statistics?

- Tool: Graffiti3 (Randy Davila) — automated conjecturing system
- Objects: Dyck paths $\leftrightarrow$ linear Nakayama algebras
- Goal: relate homological statistics to combinatorial statistics

The pipeline



Mode 3: Guided Search

The question

Can we find rare mathematical objects or sought-after examples using machine learning?

- Reinforcement learning: give the agent a quantity to minimise and let it play

Project: non-IDP simplices via RL

- **Goal:** find an s-lecture hall simplex without the Integer Decomposition Property
- **Outcome so far:** Reinforcement learning generated thousands of non-IDP simplices in dimensions 4, 5, 6, 7
    - can study properties of these simplices!

Other guided search experiments

Bijection search (OpenEvolve):

- AlphaEvolve/OpenEvolve: Instead of evolving “matrices”, evolve a block of code using an LLM
- Goal: Find bijection on Dyck paths interchanging area and bounce
  - Hard! We need new ways of framing the problem

How to use these tools

- You do not need to become an machine learning expert
  - Use LLMs! Chat about your problem, see what techniques you can use, ask for code snippets or ask them to make a skeleton folder for your experiment
- Find a well-defined mathematical question and phrase it in a way that gives you a quantity to optimise
- The timeline for doing these sorts of experiments has gone down from months to days or even hours!

Problems that seem promising

- Searching for counterexamples
- Searching for rare examples
- Searching for extremal examples
- Discovering formulas
- ...