Use of AI in mathematical research: A guide for young mathematicians

Pavel Etingof
Department of Mathematics, MIT

May 2026

Use of AI in mathematical research can be both very productive and very counterproductive. Below I offer some advice on this subject targeted at beginning mathematicians (primarily undergraduate and graduate students).[1]

1. The main principle

Mathematical researchers have a wide spectrum of attitudes toward generative AI (ChatGPT, Claude, Gemini, etc., also known as LLMs), ranging from boycotts for ethical reasons or skepticism to jubilation or fear of becoming irrelevant. But AI is now the elephant in the room that they can no longer ignore.

I propose the following philosophy:

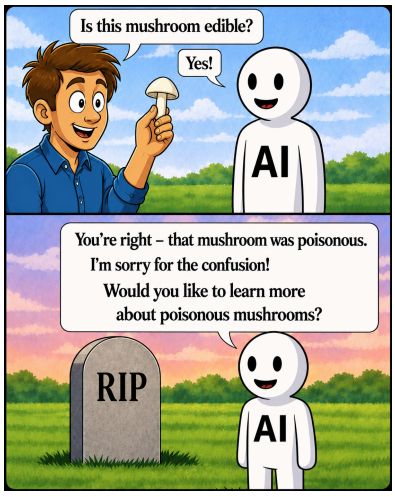
We should make the most of AI in order to know more mathematics.

This implies the main principle of using AI in mathematical research:

You should keep abreast of the math in anything you discuss with AI.

This means that if you use AI output in your paper, you have to **check and understand it in full detail**, leaving no stone unturned, and **rewrite it in your own way**, rather than copy-paste it into your writing. And you have to **be able to explain everything in your write-up on the spot.**

There are at least two reasons for this. Firstly, AI models can **hallucinate** and create good-looking but incorrect and misleading content. While they can solve difficult math problems correctly, they can also produce wrong arguments based on ridiculously nonsensical (but often tacit and well-hidden) assumptions, such as that all odd numbers are prime. This is getting better with time, but is still the case, and it can cause a fatal flaw in your paper!



[1]Given the pace of development of AI technology, this advice will go out of date in no time. I plan to update it frequently enough to keep it relevant.

Yet it is **you** who are responsible for any mistakes, however subtle, that make their way into your writing. Thus you should take any AI output very critically. Even if this output is largely correct, it is often difficult to fish out all the little imperfections and imprecisions and work it smoothly into your narrative. So it is usually more efficient and reliable to write yourself than to debug an AI-produced text.

And secondly (and most importantly), the purpose is for **you** to learn, do and enjoy mathematics. So if you relegate all these functions to AI, you defeat this purpose!

2. General advice

1. Choice of models and modes. For serious mathematical questions it is better to use a powerful version of AI. Namely, you should always use a reasoning/extended-thinking rather than the ordinary "instant" mode, and ask for a detailed, checkable solution, including lemmas, examples, and possible failure points. For the most difficult questions you may want to use a Pro mode. This may cost more tokens, but tends to produce better results. But make sure your prompt is well formulated – Pro queries are expensive! You may use a less powerful mode first and ask AI to repeat your question back to you to make sure it understood you correctly.

2. Prompting. It may be a good idea to use a sentence like "*I am a mathematics undergraduate/graduate student who is doing research on X*" as a preamble to any prompt, or put it down as your default system prompt (a message automatically taken into account within a given project). This will tell the AI model what your experience level is and help it serve you better. You may also add to this message various information about you (e.g. the books you've read and math courses you've taken). Adding sentences like "*Check your answer/proof very carefully*" to the prompt may also be helpful. AI models are hard-wired to cut corners (in an effort to save resources), and your job is not to let them do so at your expense!

3. Iteration. Experience shows that if you ask AI a difficult math question, one pass is usually not enough. You should study the output carefully, point out mistakes, and ask AI to clarify or elaborate on the points you are most interested in. Useful results are often produced only after several iterations of this process. However, avoid very long chats, especially with the more expensive models: AI may have trouble dealing with a large amount of context and also may charge tokens for re-reading it after each prompt, sharply increasing the context cost (although in many cases you can suppress this re-reading by changing the configuration parameters). Instead, ask AI to summarize the chat at a natural breaking point and start a new one, uploading the summary and asking AI to use it as a basis for answers. In general, it is a good idea to keep each chat focused on a particular topic. And don't just stare at the screen waiting for AI to respond – you can do other things in the meantime, e.g., run other chats or do your own math!

4. LaTeX. AI models know LaTeX very well, so you can ask them any LaTeX questions, or to debug your LaTeX file. You may also ask them to give any output in the form of a LaTeX file, for convenient reading.

5. Privacy. Last but not least – **you should not upload into AI any non-public material without permission from the owner.** Also, before uploading non-public data, check the platforms data-control settings and policies; for instance, in public ChatGPT, users can turn off the setting "Improve the model for everyone", in which case the data will not be used for training.

3. Learning

Learning should continue all the way through your project: mathematicians learn mathematics as long as they do mathematics. Learning consists of mastering theoretical material from books and papers and doing exercises. While some sources and exercises may be provided by your mentor or advisor, you may ask AI for more. For instance, you may use the following prompts:

1) "*What are the best sources to learn about Y?*"

2) "*What is known about the following question? (*formulate the question precisely*). Are there any references?*"

3) "*I have been reading B (*upload the file*) and on page N I have trouble understanding the proof of Theorem Z. Can you explain this proof to me in more detail?*"

4) "*Can you give me the basics about T and provide some exercises?*"

5) "*What is the motivation for the notion A?*"

6) "*How is P related to Q?*"

7) "*Can you explain the main ideas of R?*"

On the other hand, using AI to **solve** the exercises given by your mentor or advisor is **a bad idea**. The exercises are designed for you to gain technical skills needed for your research. Think of an athlete who wants to run a marathon, but for training uses a motorcycle!

In any case, in most learning environments trying to pass AI output as your exercise solution or being unable to explain your solution because you rewrote an AI solution without sufficient understanding constitutes **an inappropriate use of AI** and will be viewed as a **violation of academic integrity**.

4. Research

Using AI in your research project can be highly effective. Here is one major reason. When deciding which step to take next at any given moment (doing a computation, working out an example, etc.), you do (consciously or not) a cost-benefit analysis: utility of the step vs. its difficulty. And often you will decide that while it could be quite illuminating, it is just too taxing. But the availability of a strong AI model can dramatically lower the opportunity cost and tilt this cost-benefit analysis in favor of taking the step, which may lead to a revelation or a breakthrough in your project.

There are many good ways to use AI in research, as long as you follow the principle of "keeping abreast", acknowledge any major AI contributions in your papers and talks,

and observe privacy rules (subject to advisor expectations, coauthor consent, as well as journal and institutional policies). Namely, you may use AI to create research questions, produce data and examples, brainstorm, try to verify a statement leading to a proof or a counterexample, and help you with writing and proofreading. Here are some tips.

4.1. **Creating research questions.** You can use AI to create research questions inside your project. A good way to do this is to upload a description of your project, the most relevant papers, and your write-up to date into the AI model and use a prompt like

"*Read very carefully the attached project description, papers, and my text, and propose N research questions to consider next. For each, explain motivation, propose first steps, give references.*"

For this, you'll want to use the most powerful mode (Pro), so that AI will think for a while (it can be 20 minutes or more). While many of the questions proposed by AI may not be good, some of them can turn out to be interesting!

However, it may be hard for you to judge which of the questions are good. For this reason, before attempting any of these questions, it is better to ask your mentor, advisor, or another expert in the field. At the minimum, before working on any question, you should do a literature search (which can be efficiently done using AI, as explained in the previous subsection) to determine if the problem has already been addressed in the literature.

4.2. **Brainstorming.** You can also use AI to brainstorm for ideas how to make progress in your project. You may use a prompt like

"*I am working on the following problem. (*Formulate the problem precisely or attach a file with formulation*) Here are some of my insights. (*Share your own ideas*) Which of my ideas are most promising? Can you suggest ideas of how to proceed? Can you point out any patterns in this data? Can you find any useful references?*"

Then study the output to see if it contains any good ideas, perhaps doing several iterations as explained above. Discuss with your advisor or collaborators, then repeat. Eventually it may yield something useful!

4.3. **Generating data and examples (by Andrew Sutherland).** Frontier models like Claude, Chat-GPT, or Gemini are now quite good at producing data and analyzing it. One of the most useful things you can do at the early stages of a research project is thinking about ways to generate examples, both to help build intuition and as a way to develop and test conjectures. Even very abstract questions often have special cases or implications that can be made concrete. In the past the task of figuring out how to do this, writing the code, testing it, optimizing it, and running it might have been an entire research project in itself, but this is now something that can be done by models like Claude Opus 4.7 and GPT-5.5 Pro. The interaction that takes place while doing this can be very instructive (trying to explain to the model what you want it to do helps you understand the problem better, and you will learn things along the way).

4.4. **Proving and disproving.** You can use powerful versions of AI for proving and disproving mathematical statements. However, this should be done with care. AI models are generally good at summarizing known material and mimicking well established arguments, but not as good at producing really original mathematics. When asked to do the latter, they will often hallucinate. That's why **any AI output claiming to prove or disprove a statement whose status was previously unknown should be taken with a large**

grain of salt. You should always treat AI output as a starting point, at best an intermediate step, and never as a final result.

One danger is that instead of doing math you will be mired in checking smooth-looking but actually buggy AI-generated arguments. To avoid this, instead of reading the AI output right away, you may first feed it into another (or the same) AI model and say something like: "*This proof of statement Y was given to me by AI. I am skeptical. Can you check it?*"

Quite possibly, the second AI model will find a mistake. Then you can feed the response back into the first model and keep iterating until the process converges - to either the first model admitting it has no proof, or to both models agreeing that the proof is correct. In the latter case, you may try to check it yourself (having run it by a third AI model first, if you like). This way, you can avoid wasting your time reading low quality AI outputs.[2] Note however that agreement between two or three models is by itself **not** strong evidence of correctness, since models can share training data, styles of error, and plausibility biases.

As usual, if you are having a hard time understanding a step in the proof, you can ask AI to elaborate. But at the end you should write your own proof!

It may be helpful to ask the AI model to work out an example instead of (or at least separately from) the general case. Then the output is more likely to be useful and will be easier for you to check.

With time, you should be able to develop some intuition on when it is relatively safe to trust AI and when it is less so. But still, be on guard at all times! Remember that AI is exceptionally good at fooling humans!

As noted in the previous subsection, you may also ask AI to do computations, although for serious ones it's better to run your own code in Mathematica, Sage, and the like. However, you may ask a powerful enough version of AI (e.g. Claude Code) to write that code (possibly proposing an idea for an algorithm, perhaps after discussing it with AI). The code can, of course, contain bugs, but you can go through a few iterations to debug it; this can be faster than writing it yourself. It is also a good practice to ask AI to supplement the code with detailed comments on what is done in each part; this will make it easier for you to check if it works correctly.

Warning. You should carefully check all references used in AI-generated arguments. Remember that AI can produce nonexistent references, refer to non-existing statements in existing sources, or misunderstand what is claimed in the papers it reads. Also remember that many articles, especially old ones, are behind a paywall, and AI has no access to the full text. In this case it may make assumptions about their contents from the abstract or other sources, which may lead to misinterpretations. Your mentor or advisor should be able to help you sort this all out.

Remark (by Andrew Sutherland). At this stage, AI is often better at disproving conjectures than proving them, but this is still very useful! Generating counterexamples is one way to do this, but it's true more generally, including in formal settings, as noted in Talia Ringer's recent article on AlphaProof: `https://www.nature.com/articles/d41586-025-03585-5`

4.5. **Collaboration vs. relegation.** While AI can be very helpful in your research, for good results you should **think about the problem yourself together with AI** (in agreement with the "keeping abreast" principle), rather than just watch AI do research for you. In

[2] I learned this technique from Jesse Geneson.

other words, you should think of AI as a **powerful tool**, or perhaps a (not always reliable) **digital collaborator** possessing knowledge and abilities that you yourself lack, and **work together** to make the most of them, rather than **relegate** the process to AI altogether. Struggling with the problem, which can be short-circuited by using AI, is a vital part of research process. This is the only way to develop the understanding and intuition needed to make a breakthrough. The less experience you have, the more important this is!

In short, AI should **not** reduce your time or effort spent on mathematics; it should just increase the efficiency so that you can achieve more.

For beginning researchers, a good practice is trying a problem yourself before asking AI to solve it, or asking AI for hints rather than full solutions first. In the end, the goal is not to use you and AI as a machine to churn out as many theorems as possible, but rather to make progress on the project while also giving you training and experience in doing research. If you entrust everything to AI, how will you develop your own research skills?

After all, discovering new mathematics is the most enjoyable part of this whole endeavor. So why outsource this pleasure to a soulless chatbot? Imagine that you go to a fine restaurant and instead of savoring the delicious dishes, put them through a chewing machine and then swallow the resulting goo. How much fun would this be?

4.6. **Formal verification (by Andrew Sutherland).** In cases where it is feasible (which largely depends on whether the topic you are working on has adequate coverage in MathLib) it's reasonable to consider asking an LLM to try to formalize and verify a statement in **Lean**. If the statement is fairly simple, most of the frontier models know Lean well enough to be able to write a script you can copy and paste into an online verifier like `https://live.lean-lang.org/` or `https://axle.axiommath.ai/verify_proof`. But for anything more substantial, it's helpful to give the LLM some additional tools (e.g. a "skills" file as in `https://github.com/cameronfreer/lean4-skills` and an interface like the lean-lsp-mcp `https://github.com/oOo0oOo/lean-lsp-mcp`). There are also third-party auto-formalizers like AlphaProof, Aristotle, AxiomProver, Gauss, Godel-Prover, Kimina-Prover, ..., many of which are free to use.

Remark 4.1. LLMs make it possible for people who have never used Lean or a formal theorem prover before to leverage formal verification in a way that was not possible a year ago. But you need to be sure the LLM has correctly formalized the statement you care about. This requires some understanding of the formal system (but much less than would be needed to write the proof).

5. Writing and proofreading

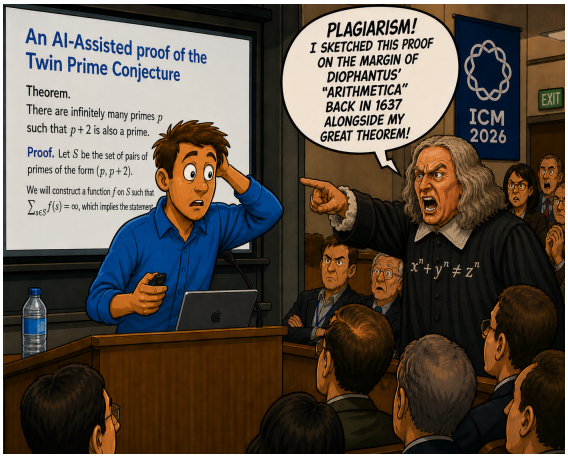
At the writing stage, you can use AI for helping you with LaTeX and with diagrams or pictures, although it can be hard to explain what you want drawn. Sometimes it is better to sketch it on paper by hand, scan with your cell phone, upload to AI and ask to generate a PNG file or a LaTeX code for the diagram.

You may also use AI for proofreading. You can upload your text and say something like: "*This is my paper. Read it very carefully and point out misprints, mistakes, mismatches of notation, English issues, etc. Read sections X.Y to Z.T. Give me comments.*" It's advisable to ask AI to read a small portion at a time (say, 10 pages), then the quality may be better. I've been using this method for my own texts and those of my students, and I strongly recommend it (e.g. ChatGPT 5.5 Extended thinking does a great job!). Of course, it does not replace good old human proofreading, but it's nice to do an AI proofreading (followed by corrections) before the human one!

AI models are generally good at catching misprints, correcting English, finding mismatched and undefined notation, but less good at finding subtle mathematical errors, although on occasion they can do that as well. However, if you want AI to check your paper seriously, you should ask it to check individual proofs, and keep adding details and clarifications until it no longer finds issues. But make sure you prompt the AI to be critical and not to please you (and make its job easy) by saying the proof is correct without thorough checking!

At the same time, copy-pasting significant chunks of unpolished AI output into your paper is **a bad idea**.[3] You should really do the writing yourself!

One more reason for this is that AI can tacitly copy something from someone else's work without reference. So if you copy-paste this output into your text and it gets publicized in that form, this may constitute **plagiarism**. And even though you'd do this inadvertently, you'd still be responsible for it!



[3]Circulating unchecked AI output among a research group is acceptable, but only if its unverified status is clear to everyone involved.

For this reason, if you plan to use its output even in paraphrased form, it is a good practice to **always ask AI to list the sources it used** and then **independently verify them** (on arXiv.org, MathSciNet, etc.) and check if a reference to any of them is in order. Also it is a good idea to aggressively request the most related literature multiple times, and to **properly cite these original works in the reference section of your article**. For instance, it's easy to just cite Google in the reference section, but no one does this. It's far better for people to list the original resources that they are using, as this helps others understand the scope of the literature. You also might appreciate some credit for your ideas when they are being used!

Note that when asked to proofread, by default some LLMs will prefer to rewrite your text themselves, instead of giving you comments. This is for many reasons very undesirable, as explained above. So you should make it clear to the LLM that it should just give you a list of comments/corrections and not try to rewrite the text for you.

Remark 5.1. [4] While proverbially, during a conversation the most social mathematicians look at their interlocutor's shoes instead of their own, mathematics is, in fact, an intrinsically social endeavor. When we say that our goal in using AI is "to know or understand more mathematics", the actual meaning is "to achieve understanding for **the mathematical community**". Historically, "a problem solved" has always meant that the community has achieved such understanding, or at least has developed an accessible path towards achieving it, and there is no reason this should change with the advent of AI.

An AI-generated proof is like a revelation beamed on us from an alien civilization. It is missing the extra work that transforms it into "understanding mathematics" - a notion whose meaning should remain invariant in the age of AI. Thus it is the job of the person publishing that argument to do that work for the community, which means integrating any new ideas into the body of common knowledge by finding ways to explain these ideas so that people find them instructive. This entails exchanging emails, having in-person discussions, giving talks, etc. Just posting the proof on the arXiv does none of that!

6. Preparing a presentation

You can use AI for preparing your presentation slides. You may upload a text and say "*Based on this paper, make and give me a Beamer file for a 12-minute talk.*"

You may add to the prompt what you want to cover, how much time you want to spend on what, etc. Of course, you may need a few iterations and the file will have to be seriously edited after that. But this can save you time by making a reasonable first draft.

7. Acknowledgments

If you used AI for creative tasks and use the results in your paper, it is a good practice to acknowledge it by saying which models you used and what exactly their roles were; in fact, it is required by many journals. Even if AI helped you with some proofs, you will have full credit for the results of the paper. But again, don't forget that this full credit comes with full responsibility for the content![5]

[4] I thank Paul Seidel for this comment.

[5] Some people even make the AI model a coauthor of their paper. This is not necessary (and disallowed by many journals), as the status of AI is different – unlike a human author, it is not responsible for correctness and originality of the content. However, explaining exactly what was done by AI is highly advisable.

8. Other resources (by Andrew Sutherland)

1. While it is now somewhat out of date, there is a pretty extensive list of AI for Math Resources that was compiled during the National Academies workshop that took place in 2023 and has been updated a few times since then. You can find it at `https://docs.google.com/document/d/1kD7H4E28656ua8jOGZ934nbH2HcBLyxcRgFDduH5iQ0`.

2. While it doesn't offer specific tips, the recent expository paper by Klowden and Tao "Mathematical methods and human thought in the age of AI" might be useful background reading as you think about how to make the most of AI in order to know more mathematics: `https://arxiv.org/abs/2603.26524`.

3. The paper `https://arxiv.org/pdf/2605.20695v1` describes a very impressive recent success of AI in math (disproof of Erdös' conjecture).

9. Some useful links

MIT guidance on AI use: `https://ist.mit.edu/ai-guidance`

MIT guidance on citing AI tools: `https://libguides.mit.edu/cite-AI-tools`

AMS Journals policy on the use of AI: `https://www.ams.org/publications/journals/policies/UseofArtificialIntelligence`

10. Checklist for responsible AI use

To summarize - before using AI in learning or research, it's good to ask yourself the following questions.

1. Understanding and responsibility.

- Do I fully understand every mathematical statement, proof, computation, or example that I am taking from AI output?
- Can I explain it clearly on the spot to my advisor, mentor, collaborator, or audience?
- Have I checked all definitions, hypotheses, edge cases, and hidden assumptions?
- Am I using AI to help me learn and do mathematics, rather than to avoid learning and doing mathematics?

2. Privacy and permissions.

- Am I allowed to upload the material I am giving to AI?
- If the material belongs to a collaborator, advisor, student, journal, conference, or institution, have I obtained permission when needed?
- Have I checked the relevant data-control settings and privacy policy of the platform I am using?
- If the material is sensitive or non-public, should I instead use a more private setting, such as an institutional account, an enterprise account, a local model, or no AI at all?

3. Learning and exercises.

- If this is an exercise assigned by a mentor, advisor, or instructor, is AI use permitted?
- Am I asking for hints, explanations, or background rather than outsourcing the solution?
- If I use an AI-generated explanation, have I reconstructed the argument myself?

4. Research ideas and proofs.

- Have I treated AI-generated research questions, conjectures, proofs, and counterexamples skeptically?
- Have I checked whether the question or result is already known?

- Have I discussed promising AI-generated ideas with my advisor, mentor, collaborators, or another expert when appropriate?

- If AI suggests a proof, have I verified it myself rather than relying on the model's confidence?

- If another AI model also agrees with the proof, do I remember that this is still not a substitute for my own verification?

5. Computations, code, and data.

- Have I checked that the code does what I asked it to do?

- Have I tested the code on simple examples where I already know the answer?

- Have I checked for off-by-one errors, missing cases, numerical instability, and incorrect assumptions?

- If the computation is important to the paper, have I saved enough code, data, and explanation to make it reproducible?

6. References and literature search.

- Have I asked AI for sources it has used?

- Have I done a thorough literature search and checked if the argument or approach suggested by AI has appeared previously?

- Have I verified every reference suggested by AI in the actual paper, book, MathSciNet, zbMATH, arXiv, or a journal website?

- Have I checked that the cited source really contains the claimed theorem, definition, example, or argument?

- Have I made sure that AI has not invented a reference, confused two papers, or inferred content from an abstract without reading the full text?

7. Writing.

- Have I avoided copy-pasting substantial AI-generated text into my paper or presentation?

- Have I rewritten any useful AI-generated material in my own voice and checked it for correctness?

- Have I made sure that the notation, terminology, and style are consistent with the rest of my text?

- Have I asked AI for comments and corrections rather than letting it silently rewrite the paper for me?

8. Acknowledgments.

- Did AI make a substantial creative contribution to the project, such as suggesting research directions, generating examples, helping with a proof, writing code, or producing figures?

- If so, have I acknowledged which model was used and what it was used for?

- Have I checked the AI policies of the journal, conference, course, institution, or funding agency?

- If there are coauthors, have we agreed on how AI use should be disclosed?

Final test:

If I removed the AI chat history, would I still understand, be able to reproduce, and be willing to take full responsibility for everything in the paper or presentation?

11. Acknowledgments

I thank Jesse Geneson, Slava Gerovitch, Tanya Khovanova, Andrew Sutherland and Chelsea Walton for useful comments. This work is supported in part by an Amazon AGI Faculty Award to Prof. Anna Rumshisky from the Miner School of Computer and Information Sciences at UMass Lowell. The images were created by ChatGPT.