How to explain advanced mathematics to a computer

Riccardo Brasca

Université Paris Cité
Institut de Mathématiques de Jussieu-Paris Rive Gauche

15th December, 2021

What is formalized mathematics?

Formalization is the process of using a computer to *reason*.

What is formalized mathematics?

Formalization is the process of using a computer to *reason*.

This is different from using tools as Sage, Maple, Mathematica...

What is formalized mathematics?

Formalization is the process of using a computer to *reason*.

This is different from using tools as Sage, Maple, Mathematica...

Formalization is done using *proof assistants*.

What is formalized mathematics?

Formalization is the process of using a computer to *reason*.

This is different from using tools as Sage, Maple, Mathematica...

Formalization is done using *proof assistants*.
There are several proof assistants: we will use Lean.

A proof assistant has two components:

A proof assistant has two components:

- The *kernel*.

A proof assistant has two components:

- The *kernel*. It checks that the proof is formally correct.

A proof assistant has two components:

- The *kernel*. It checks that the proof is formally correct.
- The *elaborator*.

A proof assistant has two components:

- The *kernel*. It checks that the proof is formally correct.

- The *elaborator*. It translates a proof written by a human to something the kernel can check.

A proof assistant has two components:

- The *kernel*. It checks that the proof is formally correct.
- The *elaborator*. It translates a proof written by a human to something the kernel can check.

Suppose we have $a, b \in \mathbb{R}$

A proof assistant has two components:

- The *kernel*. It checks that the proof is formally correct.

- The *elaborator*. It translates a proof written by a human to something the kernel can check.

Suppose we have $a, b \in \mathbb{R}$ and we write $a + b$.

A proof assistant has two components:

- The *kernel*. It checks that the proof is formally correct.

- The *elaborator*. It translates a proof written by a human to something the kernel can check.

Suppose we have $a, b \in \mathbb{R}$ and we write $a + b$. We want Lean to know that this makes sense because $\mathbb{R}$ is a group.

A proof assistant has two components:

- The *kernel*. It checks that the proof is formally correct.
- The *elaborator*. It translates a proof written by a human to something the kernel can check.

Suppose we have $a, b \in \mathbb{R}$ and we write $a + b$. We want Lean to know that this makes sense because $\mathbb{R}$ is a group. But $\mathbb{R}$ is also a ring, a field...

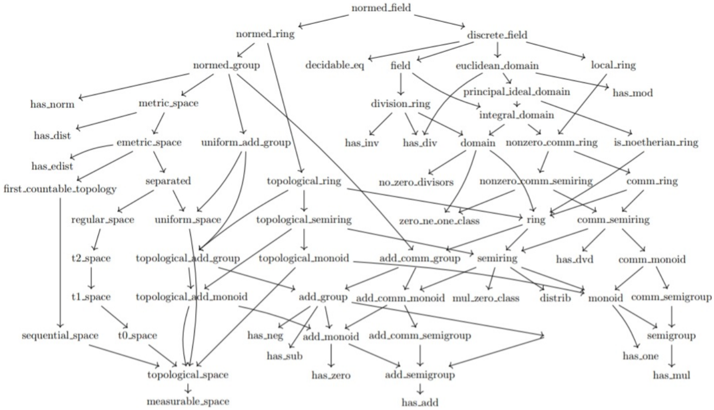

Figure: Image by Jeremy Avigad

We are having fun.

We are having fun.

Formalization is challenging.

We are having fun.

Formalization is challenging.

It invites us to rethink basic mathematical concepts from a different point of view.

Checking correctness

Checking correctness

There are proofs are too big even for experts.

Checking correctness

There are proofs are too big even for experts.

Classification of finite simple groups.

Checking correctness

There are proofs are too big even for experts.

Classification of finite simple groups.
Results in arithmetic geometry.

Checking correctness

There are proofs are too big even for experts.

Classification of finite simple groups.
Results in arithmetic geometry.

Peter Scholze asked for a verification of a very technical result in his recent work with Dustin Clausen.

Scholze on the Xena project blog:

Scholze on the Xena project blog:

Why do I want a formalization?

Scholze on the Xena project blog:

Why do I want a formalization?

- *... I think the theorem is of utmost foundational importance, so being 99.9 % sure is not enough.*

Scholze on the Xena project blog:

Why do I want a formalization?

- *... I think the theorem is of utmost foundational importance, so being 99.9 % sure is not enough.*
- *... As it will be used as a black box, a mistake in this proof could remain uncaught.*

Scholze on the Xena project blog:

Why do I want a formalization?

- *... I think the theorem is of utmost foundational importance, so being 99.9 % sure is not enough.*
- *... As it will be used as a black box, a mistake in this proof could remain uncaught.*
- *... In the end, we were able to get an argument pinned down on paper, but I think nobody else has dared to look at the details of this, and so I still have some small lingering doubts.*
- *... It is the kind of argument that needs to be closely inspected.*

Scholze on the Xena project blog:

Why do I want a formalization?

- *... I think the theorem is of utmost foundational importance, so being 99.9 % sure is not enough.*
- *... As it will be used as a black box, a mistake in this proof could remain uncaught.*
- *... In the end, we were able to get an argument pinned down on paper, but I think nobody else has dared to look at the details of this, and so I still have some small lingering doubts.*
- *... It is the kind of argument that needs to be closely inspected.*
- *While I was very happy to see many study groups on condensed mathematics throughout the world, to my knowledge all of them have stopped short of this proof. (Yes, this proof is not much fun. . .)*

- *From what I hear, it sounds like the goal is not completely out of reach. ... If achieved, it would be a strong signal that a computer verification of current research in very abstract mathematics has become possible. I'll certainly be excited to watch any progress.*

- *From what I hear, it sounds like the goal is not completely out of reach. ... If achieved, it would be a strong signal that a computer verification of current research in very abstract mathematics has become possible. I'll certainly be excited to watch any progress.*
- *I think this may be my most important theorem to date. (It does not really have any applications so far, but I'm sure this will change.) Better be sure it's correct...*

- *From what I hear, it sounds like the goal is not completely out of reach. ... If achieved, it would be a strong signal that a computer verification of current research in very abstract mathematics has become possible. I'll certainly be excited to watch any progress.*
- *I think this may be my most important theorem to date. (It does not really have any applications so far, but I'm sure this will change.) Better be sure it's correct...*

Scholze asked for a formalization of the following theorem.

Scholze asked for a formalization of the following theorem.

Theorem

Let $0 < p' < p \leq 1$ be real numbers, S a profinite set and V a p-Banach space.

Scholze asked for a formalization of the following theorem.

Theorem

Let $0 < p' < p \leq 1$ *be real numbers,* S *a profinite set and* V *a p-Banach space. We have*

$$\operatorname{Ext}^1_{\mathrm{Cond(Ab)}}(\mathcal{M}_{p'}(S), V) = 0.$$

Scholze asked for a formalization of the following theorem.

Theorem

Let $0 < p' < p \leq 1$ be real numbers, S a profinite set and V a p-Banach space. We have

$$\mathrm{Ext}^1_{\mathrm{Cond(Ab)}}(\mathcal{M}_{p'}(S), V) = 0.$$

Six months after the challenge we formalized the following proposition.

Scholze asked for a formalization of the following theorem.

Theorem

Let $0 < p' < p \leq 1$ be real numbers, S a profinite set and V a p-Banach space. We have

$$\mathrm{Ext}^1_{\mathrm{Cond(Ab)}}(\mathcal{M}_{p'}(S), V) = 0.$$

Six months after the challenge we formalized the following proposition.

Proposition

Fix radii $1 > r' > r > 0$. For any m there is some k and c_0 such that for all profinite sets S and all r-normed $\mathbb{Z}[T^{\pm 1}]$-modules V, the system of complexes

$$C^\bullet \colon \widehat{V}(\overline{\mathcal{M}}_{r'}(S)_{\leq c})^{T^{-1}} \to \widehat{V}(\overline{\mathcal{M}}_{r'}(S)^2_{\leq \kappa_1 c})^{T^{-1}} \to \cdots$$

is $\leq k$-exact in degrees $\leq m$ for $c \geq c_0$.

Proposition

Fix radii $1 > r' > r > 0$. For any m there is some k and c_0 such that for all profinite sets S and all r-normed $\mathbb{Z}[T^{\pm 1}]$-modules V, the system of complexes

$$C^\bullet \colon \widehat{V}(\overline{\mathcal{M}}_{r'}(S)_{\leq c})^{T^{-1}} \to \widehat{V}(\overline{\mathcal{M}}_{r'}(S)^2_{\leq \kappa_1 c})^{T^{-1}} \to \cdots$$

is $\leq k$-exact in degrees $\leq m$ for $c \geq c_0$.

We are now close to the end of the project.

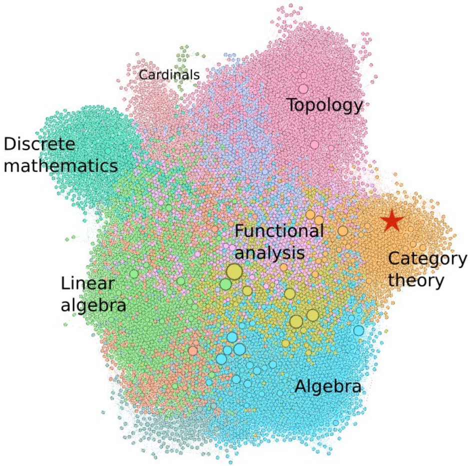

Figure: Image made by Patrick Massot

It can help the working mathematician

It can help the working mathematician

- The reader can choose the level of details.

It can help the working mathematician

- The reader can choose the level of details.

- Database of *results* rather than database of papers.

It can help the working mathematician

- The reader can choose the level of details.
- Database of *results* rather than database of papers.
- Collaboration is easier.

It can help the working mathematician

- The reader can choose the level of details.
- Database of *results* rather than database of papers.
- Collaboration is easier.
- Teaching.

Mathematical gains

Formalization can help *understanding*.

Mathematical gains

Formalization can help *understanding*.

- Consider the following theorem.

Lemma

Let (u_n) and (v_n) be sequences of real numbers and let $\ell \in \mathbb{R}$. If $\lim u_n = \ell^+$ and $\lim v_n = -\infty$ then

$$\lim(u_n + v_n) = -\infty.$$

Mathematical gains

Formalization can help *understanding*.

- Consider the following theorem.

Lemma

Let (u_n) *and* (v_n) *be sequences of real numbers and let* $\ell \in \mathbb{R}$. *If* $\lim u_n = \ell^+$ *and* $\lim v_n = -\infty$ *then*

$$\lim(u_n + v_n) = -\infty.$$

This is done in Lean (and in other proof assistants) using *filters*

Mathematical gains

Formalization can help *understanding*.

- Consider the following theorem.

Lemma

Let (u_n) *and* (v_n) *be sequences of real numbers and let* $\ell \in \mathbb{R}$. *If* $\lim u_n = \ell^+$ *and* $\lim v_n = -\infty$ *then*

$$\lim(u_n + v_n) = -\infty.$$

This is done in Lean (and in other proof assistants) using *filters*. Already done in Bourbaki.

Mathematical gains

Formalization can help *understanding*.

- Consider the following theorem.

Lemma

Let (u_n) *and* (v_n) *be sequences of real numbers and let* $\ell \in \mathbb{R}$*. If* $\lim u_n = \ell^+$ *and* $\lim v_n = -\infty$ *then*

$$\lim(u_n + v_n) = -\infty.$$

This is done in Lean (and in other proof assistants) using *filters*. Already done in Bourbaki.

- Breen-Deligne resolution in LTE.

Let's play with Lean!

Let's play with Lean!
We will prove the following results.

Let's play with Lean!
We will prove the following results.

- For all $n \in \mathbb{N}$ we have $n + 0 = n$.

Let's play with Lean!
We will prove the following results.

- For all $n \in \mathbb{N}$ we have $n + 0 = n$.
- For all $n \in \mathbb{N}$ we have $0 + n = n$.

Let's play with Lean!
We will prove the following results.

- For all $n \in \mathbb{N}$ we have $n + 0 = n$.
- For all $n \in \mathbb{N}$ we have $0 + n = n$.
- A computation that holds in any commutative ring.

Let's play with Lean!
We will prove the following results.

- For all $n \in \mathbb{N}$ we have $n + 0 = n$.
- For all $n \in \mathbb{N}$ we have $0 + n = n$.
- A computation that holds in any commutative ring.
- $1000 < 2^{10}$.

Let's play with Lean!
We will prove the following results.

- For all $n \in \mathbb{N}$ we have $n + 0 = n$.
- For all $n \in \mathbb{N}$ we have $0 + n = n$.
- A computation that holds in any commutative ring.
- $1000 < 2^{10}$.
- $47^{100000} < 79^{100000}$.

Let's play with Lean!
We will prove the following results.

- For all $n \in \mathbb{N}$ we have $n + 0 = n$.
- For all $n \in \mathbb{N}$ we have $0 + n = n$.
- A computation that holds in any commutative ring.
- $1000 < 2^{10}$.
- $47^{100000} < 79^{100000}$.
- Any commutative ring satisfies the strong rank condition.

Let's play with Lean!
We will prove the following results.

- For all $n \in \mathbb{N}$ we have $n + 0 = n$.
- For all $n \in \mathbb{N}$ we have $0 + n = n$.
- A computation that holds in any commutative ring.
- $1000 < 2^{10}$.
- $47^{100000} < 79^{100000}$.
- Any commutative ring satisfies the strong rank condition.

Definition

We say that a ring R satisfies the *strong rank condition* if the existence of an injective linear map

$$R^m \hookrightarrow R^n$$

implies $m \leq n$.

Definition

We say that a ring R satisfies the *strong rank condition* if the existence of an injective linear map

$$R^m \hookrightarrow R^n$$

implies $m \leq n$.

Theorem

Any commutative ring satisfies the strong rank condition.

Proof.

It is enough that there is no injective linear map $f\colon R^{n+1} \hookrightarrow R^n$.

Proof.

It is enough that there is no injective linear map $f\colon R^{n+1} \hookrightarrow R^n$. Let f be such a function.

Proof.

It is enough that there is no injective linear map $f\colon R^{n+1} \hookrightarrow R^n$. Let f be such a function. Let g be the composition

$$g\colon R^{n+1} \hookrightarrow R^n \hookrightarrow R^{n+1}.$$

Proof.

It is enough that there is no injective linear map $f\colon R^{n+1} \hookrightarrow R^n$. Let f be such a function. Let g be the composition

$$g\colon R^{n+1} \hookrightarrow R^n \hookrightarrow R^{n+1}.$$

Then g is injective. Let P be the minimal polynomial of g and let $a_0 \in R$ be its constant term.

Proof.

It is enough that there is no injective linear map $f\colon R^{n+1} \hookrightarrow R^n$. Let f be such a function. Let g be the composition

$$g\colon R^{n+1} \hookrightarrow R^n \hookrightarrow R^{n+1}.$$

Then g is injective. Let P be the minimal polynomial of g and let $a_0 \in R$ be its constant term.Since g is injective, $a_0 \neq 0$.

Proof.

It is enough that there is no injective linear map $f\colon R^{n+1} \hookrightarrow R^n$. Let f be such a function. Let g be the composition

$$g\colon R^{n+1} \hookrightarrow R^n \hookrightarrow R^{n+1}.$$

Then g is injective. Let P be the minimal polynomial of g and let $a_0 \in R$ be its constant term.Since g is injective, $a_0 \neq 0$.But

$$P(g)(0,\dots,1) = (0,\dots,a_0) \neq 0$$

Proof.

It is enough that there is no injective linear map $f\colon R^{n+1} \hookrightarrow R^n$. Let f be such a function. Let g be the composition

$$g\colon R^{n+1} \hookrightarrow R^n \hookrightarrow R^{n+1}.$$

Then g is injective. Let P be the minimal polynomial of g and let $a_0 \in R$ be its constant term.Since g is injective, $a_0 \neq 0$.But

$$P(g)(0, \dots, 1) = (0, \dots, a_0) \neq 0,$$

that is absurd $\square$