Lessons from the PrimeNumberTheorem+ Project And Real Analysis Game

Alex Kontorovich

RUTGERS UNIVERSITY / AWS

PrimeNumberTheorem+

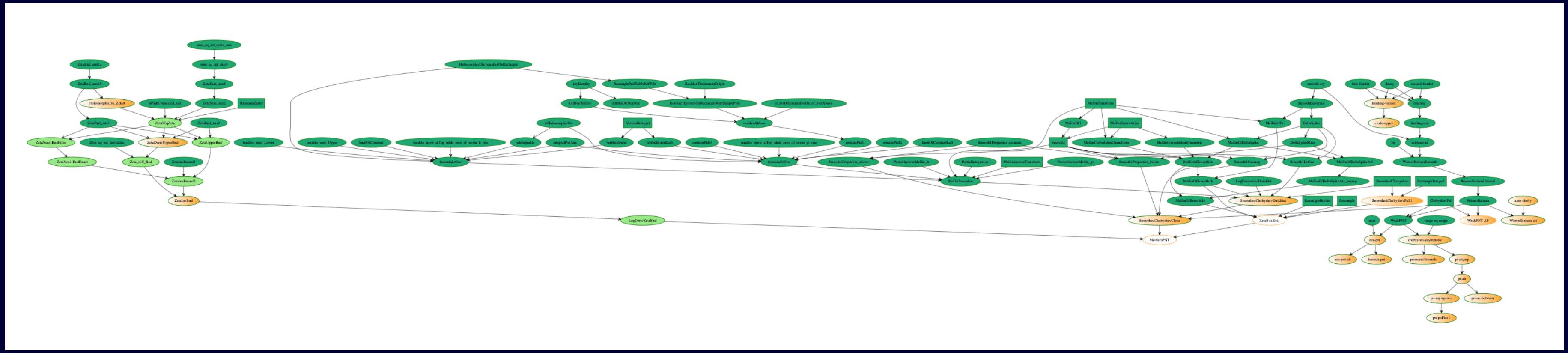
- Co-organized with Terry Tao
- Goal: Fermat will need Chebotarev Density Theorem. Special case of that is Dirichlet's theorem (primes in progressions). Didn't even have Prime Number Theorem in Mathlib. So let's get to work!

- Note: PNT has been formalized before, many times in fact.
- 2005: Avigad et al in Isabelle (Erdos-Selberg method);
- 2009: Harrison in HOL-light (Newman's proof);
- 2016: Carniero in Metamath (Erdos-Selberg);
- 2018: Eberl-Paulson in Isabelle (Newman)
- We will want to do in it a way that extends to much more general settings.

- Experiment in large scale collaboration.

PrimeNumberTheorem+

- Experiment in large scale collaboration.
- Terry had just completed leading a formalization of the Polynomial Frieman-Rusza conjecture (Gowers-Green-Manners-Tao)
- Building on many other similar projects: Sphere Eversion (Patrick Massot), Liquid Tensor Experiment, etc, etc
- Organizational infrastructure: Github + Blueprint + Zulip



PrimeNumberTheorem+

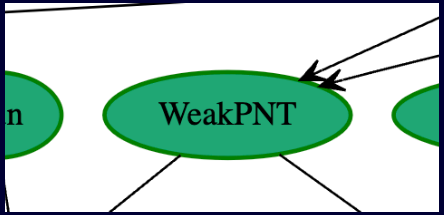
- Organizational infrastructure: Github + Blueprint + Zulip
- One “innovation” (Ian Jauslin+K): Interlace LaTeX with Lean, and scrape LaTeX before running Blueprint.
- This is amazing when coupled with Copilot, both to organize the writing, and autoformalize natural->formal language!

```
/-%%
\begin{lemma}[integrable_x_mul_Smooth1]\label{integrable_x_mul_Smooth1}
\lean{integrable_x_mul_Smooth1}\leanok
Fix a nonnegative, continuously differentiable function $F$ on $\mathbb{R}$
with support in $[1/2,2]$, and total mass one, $\int_{(0,\infty)} F(x)/x dx = 1$. Then for any
$\epsilon>0$, the function
$x \mapsto x \cdot \widetilde{1_{\epsilon}}(x)$ is integrable on $(0,\infty)$.
\end{lemma}
%%-/
lemma integrable_x_mul_Smooth1 {SmoothingF : ℝ → ℝ} (diffSmoothingF : ContDiff ℝ 1 SmoothingF)
(SmoothingFpos : ∀ x, 0 ≤ SmoothingF x)
    (suppSmoothingF : Function.support SmoothingF ⊆ Icc (1 / 2) 2) (mass_one : ∫ x in Ioi 0, Smo
    (ε : ℝ) (εpos : 0 < ε) :
    MeasureTheory.Integrable (fun x ↦ x * Smooth1 SmoothingF ε x) := by
  sorry
```

- Paul Nelson’s version in Emacs: sets TeX locally (VS extension??)

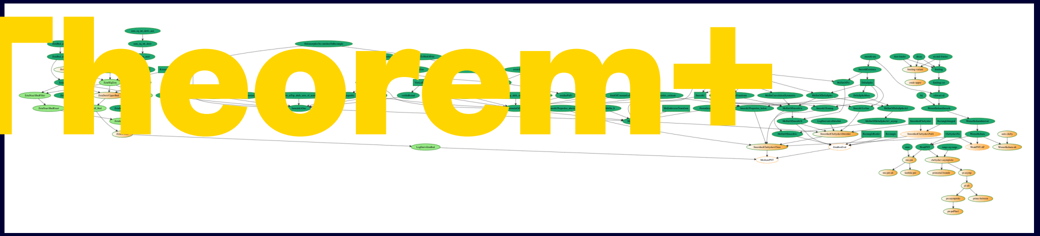
PrimeNumberTheorem+

- So, how's it going?



- Opened to the "public" on Jan 31, 2024
- Proof of PNT completed on Apr 8, 2024
- Not at all a race; we did much more than that in the meantime and since
- Phase I of the project comprised three attacks on PNT:
  - (I) using "Fourier" methods, Wiener-Ikehara Tauberian theorem. Work of Michael Stoll already reduced PNT to this.
  - (II) developing Mellin transform API (David Loeffler), pulling infinite vertical contours past poles, picking up residues. And
  - (III) Getting a "classical" error savings of $\exp(c(\log x)^{1/2})$ using Hadamard factorization (or local versions)

PrimeNumberTheorem+



- Opened to the “public” on Jan 31, 2024
- Proof of PNT completed on Apr 8, 2024

- Phase I of the project comprised three attacks on PNT:
  - (I) using “Fourier” methods, Wiener-Ikehara Tauberian theorem. Work of Michael Stoll already reduced PNT to this.
  - (II) developing Mellin transform API (David Loeffler), pulling infinite vertical contours past poles, picking up residues. And
  - (III) Getting a “classical” error savings of $\exp(c(\log x)^{1/2})$ using Hadamard factorization (or local versions)

- Mid-March 2024: Shuhao Song+Bowen Yao formalized (III) in Isabelle!
  - Built on top of much bigger Complex Analysis library…

PrimeNumberTheorem+

- Phase I of the project comprised three attacks on PNT:
  - (I) using “Fourier” methods, Wiener-Ikehara Tauberian theorem. Work of Michael Stoll already reduced PNT to this.
  - (II) developing Mellin transform API (David Loeffler), pulling infinite vertical contours past poles, picking up residues. And
  - (III) Getting a “classical” error savings of $\exp(c(\log x)^{1/2})$ using Hadamard factorization (or local versions)
- (II) Was a great excuse to get lots of students and postdocs involved in Lean, culminating in June 2025 a Simons Foundation workshop

https://leanprover-community.github.io/blog/posts/simons-lean-workshop

Three weeks later: MediumPNT was done (saving $\exp(c(\log x)^{1/10})$)

PrimeNumberTheorem+

- (II) Was a great excuse to get lots of students and postdocs involved in Lean, culminating in June 2025 a Simons Foundation workshop

https://leanprover-community.github.io/blog/posts/simons-lean-workshop

Three weeks later: MediumPNT was done (saving $\exp(c(\log x)^{1/10})$)

- The solution was a mix of: (1) large scale collaborative human formalization, organized over a dedicated zulip channel, (2) copilot speeding up writing of LaTeX+Lean, (3) AlphaProof experiments during DeepMind's visit to IAS (where I was on sabbatical).

- (III) was announced in mid-Sept by Math Inc's Gauss solver, with Jared Lichtman iteratively writing out the blueprint in more and more detail, until the solver could auto-formalize the proof.

PrimeNumberTheorem+

- The solution was a mix of: (1) large scale collaborative human formalization, organized over a dedicated zulip channel, (2) copilot speeding up writing of LaTeX+Lean, (3) AlphaProof experiments during DeepMind's visit to IAS (where I was on sabbatical).

- Compare AlphaProof solutions:

- AI invented its own language!

- Would not be easy to sort through this and figure out what's actually going on…

```
have ZetaBlowsUp : ∀ᶠ s in 𝓝[≠](1 : ℂ), ‖ζ s‖ ≥ 1 := by
  simp_all[Function.comp_def,eventually_nhdsWithin_iff,norm_eq_sqrt_real_inner]
  contrapose! h
  simp_all
  delta abs at*
  exfalso
  simp_rw [Metric.nhds_basis_ball.frequently_iff]at*
  choose! I A B using h
  choose a s using exists_seq_strictAnti_tendsto (0: ℝ)
  apply((isCompact_closedBall _ _).isSeqCompact fun and=>(A _ (s.2.1 and)).le.trans
  (s.2.2.bddAbove_range.some_mem ⟨and, rfl⟩)).elim
  use fun and ⟨a, H, S, M⟩=>absurd (tendsto_nhds_unique M (tendsto_sub_nhds_zero_iff.
  1 (( squeeze_zero_norm fun and=>le_of_lt (A _ (s.2.1 _) ) ) (s.2.2.comp S.
  tendsto_atTop)))) fun and=>?_
  norm_num[*,Function.comp_def] at M
  have:=@riemannZeta_residue_one
  use one_ne_zero (tendsto_nhds_unique (this.comp (tendsto_nhdsWithin_iff.2 ⟨ M,.
  of_forall (by norm_num[*])⟩)) ( squeeze_zero_norm ?_ ((M.sub_const 1).norm.trans
  (by rw [sub_self,norm_zero]))))
  use fun and =>.trans (norm_mul_le_of_le ↑(le_rfl) (Complex.norm_def _▸Real.
  sqrt_le_one.mpr (B ↑_ (s.2.1 ↑_)).right.le)) (by rw [mul_one])
```

PrimeNumberTheorem+

- The solution was a mix of: (1) large scale collaborative human formalization, organized over a dedicated zulip channel, (2) copilot speeding up writing of LaTeX+Lean, (3) AlphaProof experiments during DeepMind's visit to IAS (where I was on sabbatical).

- Compare AlphaProof solutions:

- To Gauss (and GoedelProver, others…)

- Clear proof outline, complete with comments

```lean
lemma lem_m_rho_is_nat (R R1 : ℝ) (hR1_pos : 0 < R1) (hR1_lt_R : R1 < R) (f : ℂ → ℂ)
    (h_f_analytic : ∀ z ∈ Metric.closedBall (0 : ℂ) 1, AnalyticAt ℂ f z)
    (h_f_nonzero_at_zero : f 0 ≠ 0)
    (hR_lt_1 : R < 1) :
    ∀ (ρ : ℂ) (h_rho_in_KfR1 : ρ ∈ zerosetKfR R1 (by linarith) f),
    analyticOrderAt f ρ ≠ ⊤ := by
  intro ρ h_rho_in_KfR1
  -- ρ lies in the closed ball of radius R1
  have hρ_closed_R1 : ρ ∈ Metric.closedBall (0 : ℂ) R1 := h_rho_in_KfR1.1
  -- R1 < R and R < 1 implies R1 < 1
  have hR1_le_R : R1 ≤ R := by linarith
  have hR1_lt_one : R1 < 1 := by linarith
  -- Hence ρ ∈ ball 0 1
  have hρ_ball1 : ρ ∈ Metric.ball (0 : ℂ) 1 := by
    have hdist_le : dist ρ (0 : ℂ) ≤ R1 := (Metric.mem_closedBall.mp hρ_closed_R1)
    have hdist_lt : dist ρ (0 : ℂ) < 1 := by linarith
    simpa [Metric.mem_ball] using hdist_lt
  -- f is analytic at ρ
```

PrimeNumberTheorem+

- Early phase: all human collaboration over zulip,
- Later: significant chunks can be done by automation (AlphaProof)

- Either way, project was a great excuse to do more analysis in Mathlib
- We didn't have Fourier inversion (now we do, Sebastian Gouzel)
- We didn't have that Fourier transform of Schwartz function is Schwartz, now we do (Gouzel + K-Loeffler-Macbeth + Beffara)

- We were also missing one of the least developed late undergrad / early grad areas of Mathlib (needed for lots of analytic number theory), namely: Complex Analysis

Complex Analysis

- I asked Claude for the core results, and which were in Mathlib:

Complex Analysis Theorems in Mathlib4

PROVED IN MATHLIB4 ✅

- Cauchy's integral theorem
- Cauchy's integral formula
- Liouville's theorem
- Fundamental theorem of algebra
- Maximum modulus principle
- Open mapping theorem
- Schwarz lemma
- Hadamard's theorem (three-lines theorem)

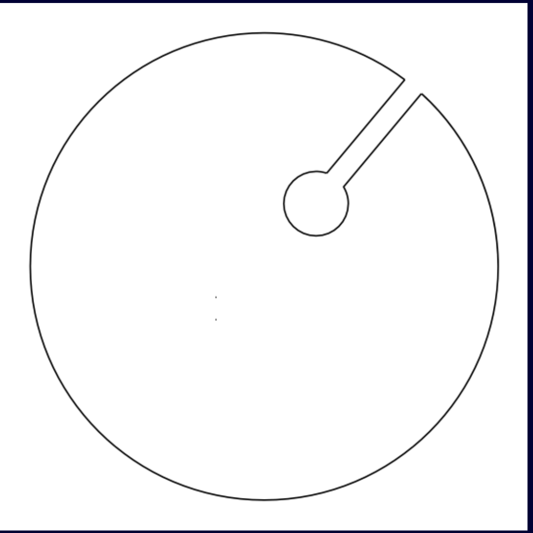
NOT YET PROVED IN MATHLIB4 ❌

- Cauchy-Riemann equations
- Morera's theorem
- Identity theorem
- Argument principle
- Rouché's theorem
- Phragmén-Lindelöf principle
- Laurent series expansion
- Residue theorem
- Riemann mapping theorem
- Weierstrass factorization theorem
- Mittag-Leffler theorem
- Picard's theorems (little and great)
- Schwarz-Christoffel formula
- Complete Möbius transformations theory

V. Beffara

Complex Analysis

- Residue Theorem:
- Don't need Jordan Curve Theorem, can do almost anything you want using explicit "Keyhole" contours
    - Do you really want to code up what this is, making sure to precompose every component with smoothing functions to stop at corners???



- Big Idea: Can do all the Complex Analysis we need just using Rectangles!
- Test (Ian Jauslin-K): holomorphic functions on disk have primitives

Complex Analysis

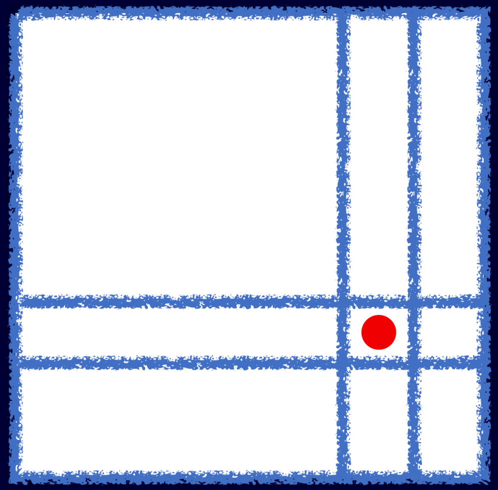
- Big Idea: Can do all the Complex Analysis we need just using Rectangles!
- We have Green's Theorem in Mathlib (Yury Kudryashov), so the integral of a holomorphic function over a rectangle is zero.

```
/-%%
\begin{definition}\label{Rectangle}\lean{Rectangle}\leanok
A Rectangle has corners $z$ and $w \in \C$.
\end{definition}
%%-/
/-- A `Rectangle` has corners `z` and `w`. -/
def Rectangle (z w : ℂ) : Set ℂ := [[z.re, w.re]] ×ℂ [[z.im, w.im]]
```

```
theorem HolomorphicOn.vanishesOnRectangle [CompleteSpace E] {U : Set ℂ}
    (f_holo : HolomorphicOn f U) (hU : Rectangle z w ⊆ U) :
    RectangleIntegral f z w = 0 :=
  integral_boundary_rect_eq_zero_of_differentiableOn f z w (f_holo.mono hU)
```

Complex Analysis

- Big Idea: Can do all the Complex Analysis we need just using Rectangles!
- We have Green's Theorem in Mathlib (Yury Kudryashov), so the integral of a holomorphic function over a rectangle is zero.
- And rectangles tile rectangles! (Not so with disks!)



- Chop big rectangle into 9 smaller rectangles
- 8 of them have integral zero!
- So you can zoom in as much as needed to a neighborhood of a pole

Complex Analysis

- So you can zoom in as much as needed to a neighborhood of a pole 
- Evaluating this boils down to: $$\int_{-\epsilon-i\epsilon}^{\epsilon+i\epsilon} \frac{ds}{s} = \int_{-\epsilon}^{\epsilon} \frac{dx}{x-i\epsilon} + \cdots$$
- Looks complicated! Need complex log??
- No, just add opposite sides together! Here is top + bottom:

$$\int_{-\epsilon}^{\epsilon} \frac{dx}{x-i\epsilon} - \int_{-\epsilon}^{\epsilon} \frac{dx}{x+i\epsilon} = 2i\int_{-1}^{1} \frac{dx}{x^2+1} = 2i(\arctan 1 - \arctan(-1)) = \pi i$$

- Same with left side + right side.
- What about pulling contours?

Complex Analysis

- What about pulling contours?
- Need things like Mellin transforms/inverses, e.g., Perron's formula:

$$\frac{1}{2\pi i}\int_{(2)} \frac{x^s}{s(s+1)} ds = \begin{cases} 0, & \text{if } x < 1 \\ 1 - \frac{1}{x}, & \text{if } x > 1 \end{cases}$$

- Human pf: if *x* < 1, pull contour to the left. Else pull right and pick up poles at *s*=0 and *s*=-1.
- Lean:

```
theorem Perron.formulaGtOne
        {x : ℝ} {σ : ℝ} (x_gt_one : 1 < x) (σ_pos : 0 < σ) :
    VerticalIntegral' (fun (s : ℂ) => ↑x ^ s / (s * (s + 1))) σ =
     1 - 1 / ↑x
```

- Pulling contours is adding/subtracting rectangles! + Limits

Complex Analysis

- What about pulling contours?
- Need things like Mellin transforms/inverses, e.g., Perron's formula:

$$\frac{1}{2\pi i}\int_{(2)} \frac{x^s}{s(s+1)} ds = \begin{cases} 0, & \text{if } x < 1 \\ 1 - \frac{1}{x}, & \text{if } x > 1 \end{cases}$$

- Systematically use smoothing! (Standard in analytic number theory *except* for PNT!...) Compare to classical "unsmoothed" approach:

Lemma 11. *For any $T > 0$ and $b > 0$ we have*

$$\begin{cases} \left| \frac{1}{2\pi i} \int_{b-iT}^{b+iT} \frac{a^s}{s} ds - 1 \right| \leqslant \frac{a^b}{\pi T |\log a|}, & a > 1 \\ \left| \frac{1}{2\pi i} \int_{b-iT}^{b+iT} \frac{a^s}{s} ds \right| \leqslant \frac{a^b}{\pi T |\log a|}. & 0 < a < 1 \end{cases}$$

PNT+

- Upshot:
  - Started massively collaborative;
  - Still very good mechanism for teaching newcomers formalization / Lean / Mathlib (slower, but speed isn't the point);
  - Newer paradigm: with API for Aristotle / AlphaProof / Gauss / Axiom / Aleph / GoedelProver / etc etc coming online, it's practically back to a single mathematician working "alone"!

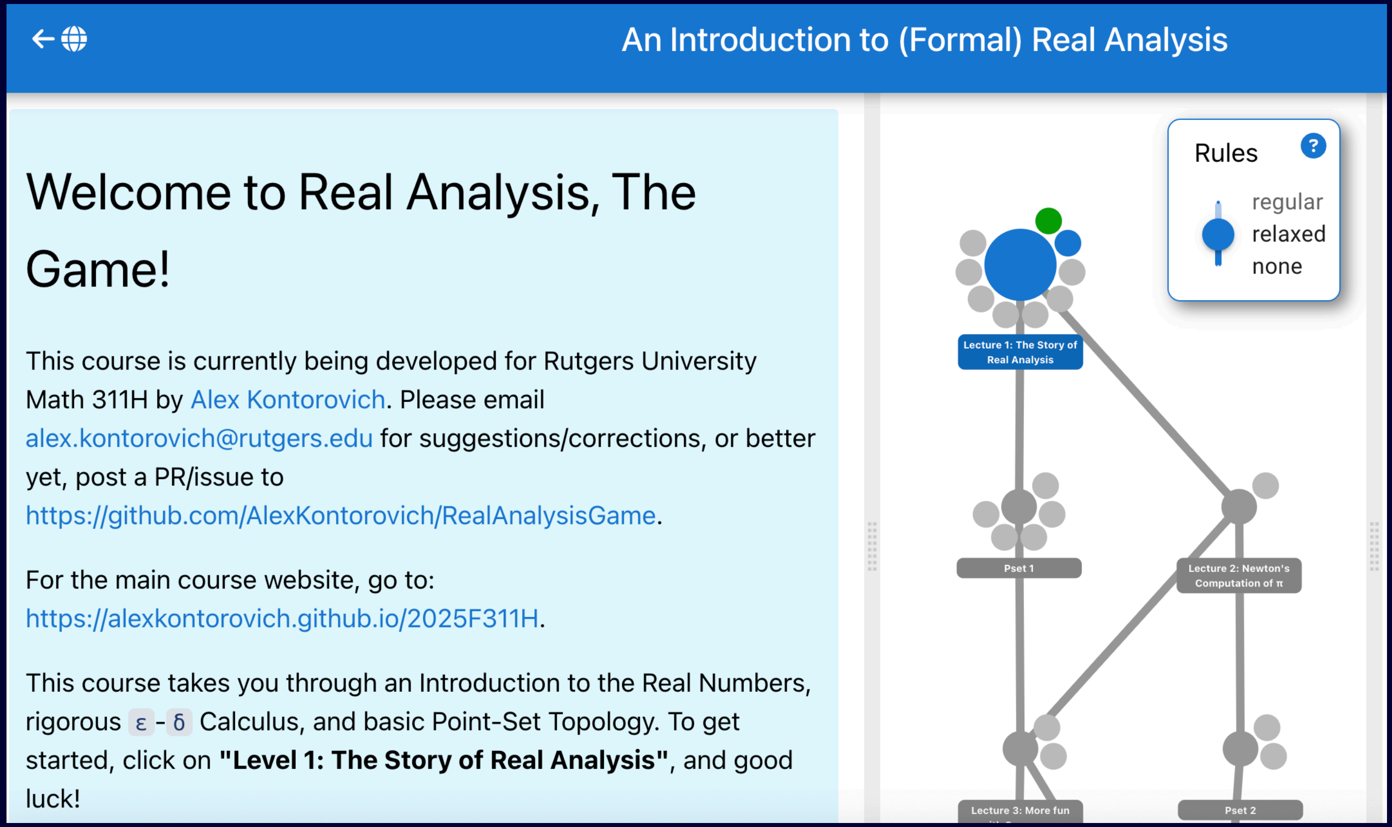
- Lots of "novel" bits of pure mathematics discovered along the way; these should make their way into "standard" textbooks, as they actually give better / simpler / cleaner proofs!

- Does this "count" as research?…

Real Analysis, The Game!

- Lots of “novel” bits of pure mathematics discovered along the way; these should make their way into “standard” textbooks, as they actually give better / simpler / cleaner proofs!

- 44 worlds, 138 levels
- Proves all “greatest hits”:
  - “Bolzano-Weierstrass”
  - “Heine-Borel”



Real Analysis, The Game!

- “Bolzano-Weierstrass”
    - Classical statement: a bounded sequence of real numbers has a convergence subsequence.
  - Classical proof: Say bounded in [0,1]; infinitely many points are in (at least) one of [0,1/2],[1/2,1]. Keep bisecting, keeping a branch where there are infinitely many. Done.
  - Not so easy (not impossible…) to formalize!

Real Analysis, The Game!

- "Bolzano-Weierstrass"
  - Classical statement: a bounded sequence of real numbers has a convergence subsequence.
  - "Better" statement: a bounded sequence of real or *rational* (!) numbers has a subsequence that is Cauchy!
  - This can be proved *before* constructing the real numbers.
  - Proof is much more elementary: (1) Every sequence has a Monotone or Antitone subsequence (consider "Peaks" and whether there are infinitely many or not). (2) Bounded + Monotone/Antitone => Cauchy! Very easy proofs.

Real Analysis, The Game!

- This can be proved *before* constructing the real numbers.
- Pedagogical philosophy: we are **not** building a house; we're thinking about **why** houses are built the way they are!
- When building a house, you pour the foundation, then frame everything, then run pipes/electric, only then do you put up the walls.
- But you *live* in the walls! So to *learn* about a house, you might make the walls first, since they're the most familiar. Then you realize you'd like running water, so you add plumbing. Walls start to fall, so you add framing. House sags, so you add foundation.

Real Analysis, The Game!

- This can be proved *before* constructing the real numbers.
- But you *live* in the walls! So to *learn* about a house, you might make the walls first, since they're the most familiar. Then you realize you'd like running water, so you add plumbing. Walls start to fall, so you add framing. House sags, so you add foundation.
- You would not **live** in such a house, but you would learn a lot from the process, about *why* everything is being done the way it is!
- Same with The Game: ideas are only introduced at the moment they're necessary.

Real Analysis, The Game!

- Same with The Game: ideas are only introduced at the moment they're necessary.

- Compare to standard Real Analysis texts: they feel like they're making a version of Mathlib:
  - Ch 1: Construct Reals (otherwise, how do you know what you're talking about?)
  - Ch 2/3: Sequences/Series
  - Ch 4: Topology in Reals (I thought this course was Analysis?)
  - Ch 5/6/7: Continuous Functions/Limits/Derivatives/Integrals

Real Analysis, The Game!

- In The Game:
  - Ch 1/2: Sequences/Series
  - (Pretend you "know" $\mathbb{R}$!)
  - Ch 3: Construct Reals!
  - Ch 4/5: Continuous Functions/Limits/Derivatives/Integrals
  - Q: When does sequence of Riemann Sums converge? A: When f is uniformly continuous! When is that? When f is continuous on a compact domain. What's that?
  - Ch 6: Topology in Reals.
  - Ch 7: Fancy things, interchange integrals+limits, etc.

- Compare to standard Real Analysis texts: they feel like they're making a version of Mathlib:
  - Ch 1: Construct Reals (otherwise, how do you know what you're talking about?)
  - Ch 2/3: Sequences/Series
  - Ch 4: Topology in Reals (I thought this course was Analysis?)
  - Ch 5/6/7: Continuous Functions/Limits/Derivatives/Integrals

Real Analysis, The Game!

- In The Game:
  - "Heine-Borel":

```
def IsCompact (S : Set ℝ) : Prop :=
  ∀ (ι : Type) (xs : ι → ℝ) (rs : ι → ℝ), (∀ i, 0 < rs i)
  → (S ⊆ ⋃ i, Ball (xs i) (rs i)) →
    ∃ (V : Finset ι), S ⊆ ⋃ i ∈ V, Ball (xs i) (rs i)
```

  - Compactness: any cover *by balls* has a finite subcover!
  - Equivalent to open covers, but much easier to work with formally.

Real Analysis, The Game!

- In The Game:
  - "Heine-Borel":

```
def IsCompact (S : Set ℝ) : Prop :=
  ∀ (ι : Type) (xs : ι → ℝ) (rs : ι → ℝ), (∀ i, 0 < rs i)
  → (S ⊆ ∪ i, Ball (xs i) (rs i)) →
    ∃ (V : Finset ι), S ⊆ ∪ i ∈ V, Ball (xs i) (rs i)
```

```
Statement IsCompact_of_ClosedSubset {S T : Set ℝ} (hST : S ⊆ T)
(hT : IsCompact T) (hS : IsClosed S) : IsCompact S := by
  intro ι xs rs rspos hcover
  change IsOpen Sᶜ at hS
  change ∀ x ∈ Sᶜ, ∃ r > 0, Ball x r ⊆ Sᶜ at hS
  choose δs δspos hδs using hS
  let Sbar : Set ℝ := Sᶜ
  let J : Type := Sbar
  let U : Type := ι ⊕ J
  let xs' : U → ℝ := fun i ↦
    match i with
    | Sum.inl j => xs j
    | Sum.inr x => x
  let rs' : U → ℝ := fun i ↦
    match i with
    | Sum.inl j => rs j
    | Sum.inr x => δs x.1 x.2
```

- Easy: compact => bounded
- Medium: compact => closed
- Medium: [a, b] => compact
- Hardest: closed ⊆ cpt => cpt
- Q: why Types? Can't we just index everything by ℕ? :)

Real Analysis, The Game!

- In The Game:
  - Lots more such “innovations”.
  - (Riemann Rearrangement / Conditional Convergence Thm…)
- The real question is: do the students actually learn Real Analysis??
- Null hypothesis: Students with no prior exposure to Lean nor to the subject matter cannot effectively learn both.
- I must reject the null hypothesis…

Real Analysis, The Game!

- I must reject the null hypothesis…

6. (10 pts) We say that a sequence $a : \mathbb{N} \to \mathbb{R}$ **diverges to infinity** if it eventually exceeds any bound. More precisely:

```
def DivToInf (a: ℕ → ℝ) : Prop := ∀ M > 0, ∃ N, ∀ n ≥ N, a n > M
```

Give a formal proof of this fact, followed by a natural language statement (explaining what the theorem says) and natural language proof:

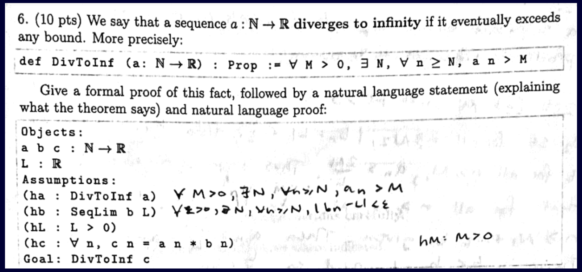
```
Objects:
a b c : ℕ → ℝ
L : ℝ
Assumptions:
(ha : DivToInf a)   ∀ M>0, ∃N, ∀n≥N, aₙ > M
(hb : SeqLim b L)   ∀ε>0, ∃N, ∀n≥N, |bₙ - L| < ε
(hL : L > 0)
(hc : ∀ n, c n = a n * b n)                   hM: M>0
Goal: DivToInf c
```

```
intro M hM
specialize hb (L/2) (by bound)
specialize ha (2M/L) (by bound)
choose N₁ hN₁ using ha
choose N₂ hN₂ using hb
specialize hN₁ (N₁+ ...) (by bound)
specialize hN₂ (...n...) (by bound)
rewrite [abs_lt] at hN₂
have f1 : L/2 < b (...n...) := by apply hN2.2; by
use N₁+N₂
intro n hn
rewrite [hc]
... aₙ·bₙ > aₙ·L/2 := by bound
```

- Two midterms and one final exam, in class, timed, on **paper**!
- Mostly natural language, with a few Lean proofs thrown in.

Real Analysis, The Game!

- I must reject the null hypothesis…

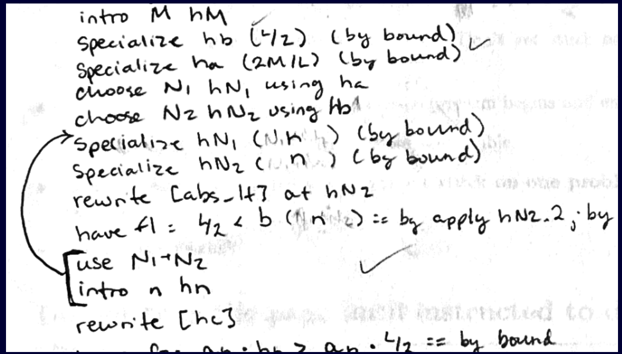


6. (10 pts) We say that a sequence $a : \mathbb{N} \to \mathbb{R}$ **diverges to infinity** if it eventually exceeds any bound. More precisely:

```
def DivToInf (a: ℕ → ℝ) : Prop := ∀ M > 0, ∃ N, ∀ n ≥ N, a n > M
```

Give a formal proof of this fact, followed by a natural language statement (explaining what the theorem says) and natural language proof:

```
Objects:
a b c : ℕ → ℝ
L : ℝ
Assumptions:
(ha : DivToInf a)   ∀ M>0, ∃N, ∀n≥N, an > M
(hb : SeqLim b L)   ∀ε>0, ∃N, ∀n≥N, |bn - L| < ε
(hL : L > 0)
(hc : ∀ n, c n = a n * b n)                hM: M>0
Goal: DivToInf c
```



```
intro M hM
specialize hb (L/2) (by bound)
specialize ha (2M/L) (by bound)
choose N1 hN1 using ha
choose N2 hN2 using hb
specialize hN1 (n) (by bound)
specialize hN2 (n) (by bound)
rewrite [abs_lt] at hN2
have f1 : L/2 < b (n) := by apply hN2.2; by
use N1+N2
intro n hn
rewrite [hc]
... an · bn > an · L/2 := by bound
```

- Two midterms and one final exam, in class, timed, on **paper**!
- Mostly natural language, with a few Lean proofs thrown in.

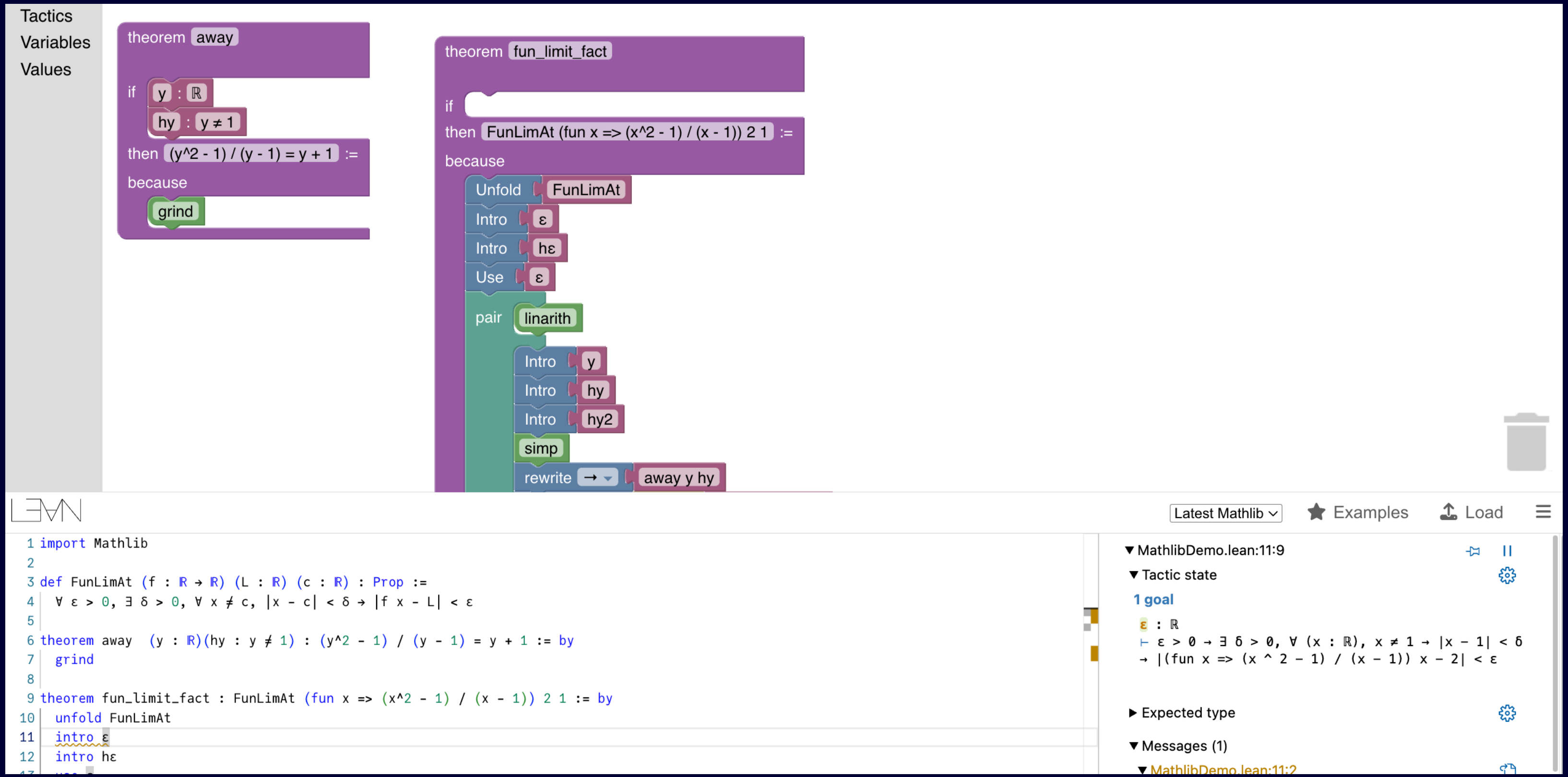
- Towards the end, many *preferred* Lean proofs to the imprecision of natural language proofs! In the future: we will name all our *hypotheses*, and look back at math with unnamed hypotheses the same way we now look at 16th century algebra!

Real Analysis, The Game!

- Towards the end, many preferred Lean proofs to the imprecision of natural language proofs! In the future: we will name all our *hypotheses*, and look back at math with unnamed hypotheses the same way we now look at 16th century algebra!
- Compare:
  - "You have some unknown quantity. When you create a second copy of this quantity and combine it with a known amount, then multiply this total by your original unknown quantity, you get a specified result"
- To:
  - $x(x+a)=b$.

Real Analysis, The Game!

- A peek into the future:…



- Real Analysis will be a Scratch-like game, with ~150 “escape room” puzzles that 12 year olds will compete to master… :)