

# 実験数学を教育から研究までやってみる

国立情報学研究所 横山 重俊

Shigetoshi Yokoyama, National Institute of Informatics

## 1 はじめに

実験数学 [1] についての適用事例は、近年クラウドコンピューティングをはじめとする情報基盤の普及とともに、さらに広がりを見せている [2, 3, 4, 5].

筆者は実験数学の実行環境として、国立情報学研究所が中心となって開発が進んでいる CoursewareHub[6, 7] と呼ばれる Jupyter Notebook 実行基盤を活用して計算機実験による教育実践を行なって来た [8, 9, 10].

これまで筆者は、群馬大学で実施した講義「コンピュータネットワークとセキュリティ」、「セキュリティ特論」の中で、暗号技術の基礎を学ぶ教育実践について、筆者が定めた実験数学の枠組みに沿った形で、実習的実験より始め、発見的実験、検証的実験という各段階を経由して定理の証明に至る部分に焦点をあて報告した [11, 12].

さらに、より研究に近い実験数学、すなわち「もっとやってみる」の領域を対象とした実験数学の事例についても報告した [13]. 研究に近い領域と設定するため、必然的に実験数学環境も研究領域で用いられているものに類似したものを用意する必要がある。研究領域で用いられている実験環境の共有を目指した取り組みには、実験で用いられたソフトウェアやデータの共有 [14], また実験の成果であるデータの共有 [15] など様々なものが存在している。

一方、データ中心型の研究手法が普及している中、オープンサイエンスの推進が叫ばれ、研究の再現性の確保が課題となっている。そのため研究データや研究ソフトウェアのオープン化と共に、その研究データを使った実験の再現性確保策として、実験を Jupyter Notebook などにより実行可能なドキュメントとして保存する提案がなされている [16].

オープンサイエンス推進の目的の中には、研究成果の教育への展開が含まれている。数学教育においても、再現性と共に提供される研究実験環境を教育目的で活用する時代がやってくると考えることができる。筆者は、数学分野における既存のデータやソフトウェアの共有を目指す取り組みを参考にしつつ、数学分野以外も対象に進むオープンサイエンスの推進の枠組みを実験数学環境構築へ適用することを試行している。

先行研究で報告した、「教育での実験数学」と「研究での実験数学」の経験から、この試行の一つとして、教育と研究という二つの領域を一続きの実験数学として連結するための取り組みを始めたので、その取り組みの枠組みとこれまでの実践内容について報告する。

## 2 背景

まず研究では、オープンサイエンスの推進のために研究データ公開やその研究データを処理するためのソフトウェアおよびソフトウェアの実行環境に関する情報の流通が議論されている。国際的にも実際それらの議論をもとにして研究管理基盤の構築が進んでいる。

数学の分野においても、例えば arXiv に preprint を投稿すると同時に、関連する実験のコードやデータを github などに置き Papers With Code というサービスで論文とそれらをリンクすることが始まっている。図 1 にその事例を示す。この事例では、arXiv の登録されたヒルベルトの第 12 問題に関連した論文 [17] の中で参照されている事実を確認するための Sage で記述されたプログラムが Papers With Code [18] 経由で参照できる。

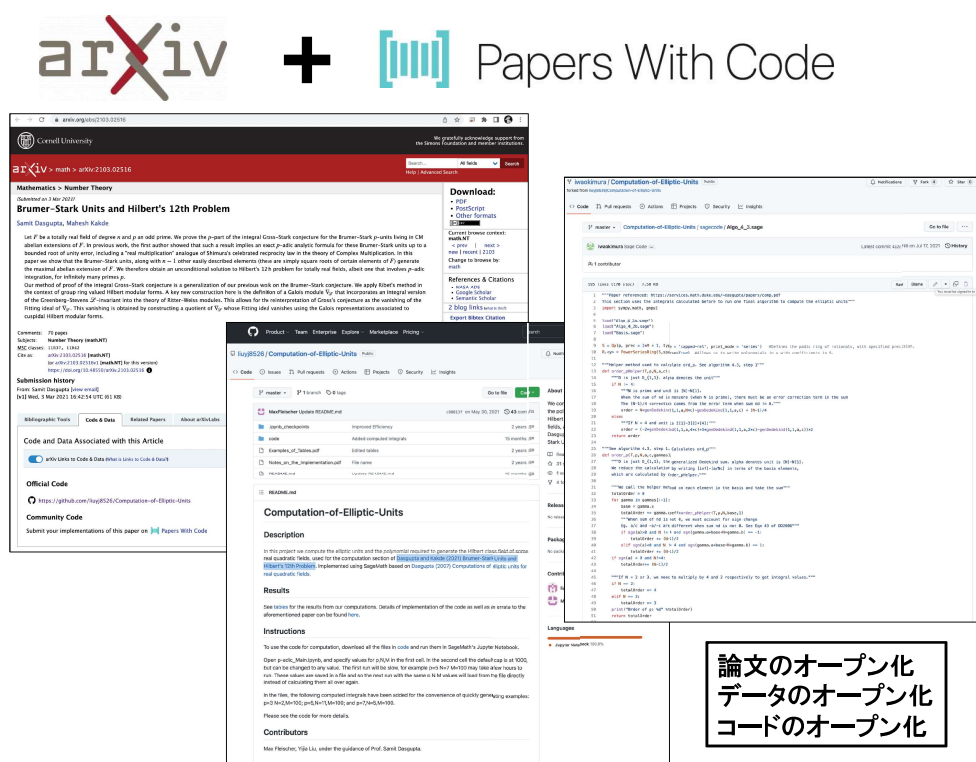
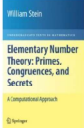


図 1: arXiv と Papers With Code の組み合わせによるオープンサイエンス推進例

一方教育においても、オープンエデュケーションの流れとして Open Educational Resources (OER) や Massive Open Online Courses (MOOCs) などの推進に向けた活動が続いている。図 2 にその事例を示す。この事例では、初等整数論のオンライン教科書 “Elementary Number Theory: Primes, Congruences, and Secrets”, William Stein [19] と連動して、例えば素数分布に関する実験を行うための Python で記述されたプログラムなどが公開されている [20]。

**Elementary Number Theory:  
Primes, Congruences, and Secrets**

William Stein  
Updated: January 2017



This is a textbook about classical elementary number theory and elliptic curves. The first part discusses elementary topics such as primes, factorization, continued fractions, and quadratic forms. In the context of cryptography, computation, and deep open research problems. The second part is about elliptic curves, their applications to algorithmic problems, and their connections with problems in number theory such as Fermat's Last Theorem, the Conjecture Number Problem, and the Conjecture of Birch and Swinnerton-Dyer. The intended audience of this book is an undergraduate with some familiarity with basic abstract algebra, e.g. rings, fields, and finite abelian groups.

On November 2008, this book was published by Springer-Verlag and can be purchased from Amazon.com. Springer-Verlag has also very generously agreed to let me make this book completely free online. So please feel free to download it!

Free Legal PDF of this book

In 2017, Pyung-Lyun Kang translated the complete book to Korean:  
Free Legal Korean Version

Also, see [github](#) for the book's LaTeX source code and newer versions.

Of course, I would greatly appreciate it if you support the book by buying it.

This book is based upon work supported by the National Science Foundation under Grant No. 0653968. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

**Question 1.2.8.** How many natural numbers  $\leq x$  are prime?

Let  $\pi(x) = \#\{p \in \mathbb{N} : p \leq x \text{ is a prime}\}$ .

For example,  $\pi(6) = \#\{2, 3, 5\} = 3$ .

Some values of  $\pi(x)$  are given in Table 1.1, and Figures 1.1 and 1.2 contain graphs of  $\pi(x)$ . These graphs look like straight lines, which may be bend down slightly.

*SAGE Example 1.2.9.* To compute  $\pi(x)$  in Sage use the `prime_pi(x)` command:

```
sage: prime_pi(6)
3
sage: prime_pi(100)
25
sage: prime_pi(3000000)
216816
```

We can also draw a plot of  $\pi(x)$  using the `plot` command:

```
sage: plot(prime_pi, 1, 1000, rgbcolor=(0, 0, 1))
```

Gauss was an inveterate computer: he wrote in an 1849 letter that there are 216,745 primes less than 3,000,000 (this is wrong but close; the correct count is 216,816).

TABLE 1.1. Values of  $\pi(x)$

$x$	100	200	300	400	500	600	700	800	900	1000
$\pi(x)$	25	46	62	78	95	109	125	139	154	168

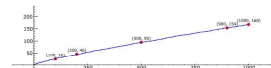


FIGURE 1.1. Graph of  $\pi(x)$  for  $x < 1000$

Gauss conjectured the following asymptotic formula for  $\pi(x)$ , which was later proved independently by Hadamard and Vallée Poussin in 1896 (but will not be proved in this book).

**Theorem 1.2.10 (Prime Number Theorem).** The function  $\pi(x)$  is asymptotic to  $x/\log(x)$ , in the sense that

$$\lim_{x \rightarrow \infty} \frac{\pi(x)}{x/\log(x)} = 1.$$

We do nothing more here than motivate this deep theorem with a few further observations. The theorem implies that

$$\lim_{x \rightarrow \infty} \frac{\pi(x)}{x} = \lim_{x \rightarrow \infty} \frac{1}{x \log(x)} = 0,$$

so for any  $\alpha$ ,

$$\lim_{x \rightarrow \infty} \frac{\pi(x)}{x/(\log(x) - \alpha)} = \lim_{x \rightarrow \infty} \frac{x/\log(x)}{x/(\log(x) - \alpha)} = \frac{\alpha}{\log(x) - \alpha} \rightarrow 1.$$

Thus  $x/(\log(x) - \alpha)$  is also asymptotic to  $\pi(x)$  for any  $\alpha$ . See [CP01, §1.1.5] for a discussion of why  $\alpha = 1$  is the best choice. Table 1.2 compares  $\pi(x)$  and  $x/(\log(x) - 1)$  for several  $x < 10000$ .

The record for counting primes is

$$\pi(10^{23}) = 1925320391606803968923.$$

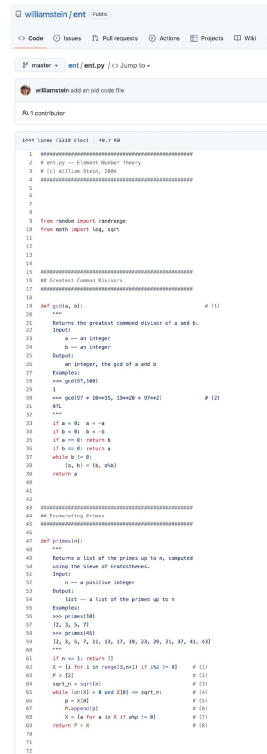


図 2: オンラインテキストと github によるオープンエデュケーション推進例

### 3 課題

研究，教育ともにオープン化が活発に議論され，いずれの領域においても官民学で連携してオープン化を推進しようとしている。しかしながら，研究と教育，それぞれの領域で別々にオープン化を推進していて，筆者が知る限り，領域間の連携についての関心がまだ低い状況にある。従来の研究と教育は図3に示すように緩やかな連携を持っていた。人は教科書などで学び，その延長として論文で既存の研究を学び，その後自らの新しい研究を行う。また，研究の成果は緩やかに教科書など教育資源に反映される。

研究と教育のオープン化を推進する際に，それらの間の連携を積極的に視野に入れた活動をして行く方法を見つけることを課題として設定した。

### 4 解決策

今後，研究と教育のオープン化が本格化すれば，それぞれのオープン化を支えるためのプラットフォームが普及すると同時に，その上で生成・活用される資源・コンテンツも増加する。これらを教育と研究をまたいで活用するためには，それらの普及が本格化する前に，まず両者のプラットフォームを図4の例のように共通化できる部分を戦略的に増やす必要がある。

その上でさらに，教育のオープン化で作成されるコンテンツと研究のオープン化で作成されるコンテンツをできるだけ関連づける。このことで教育から研究への移行がス

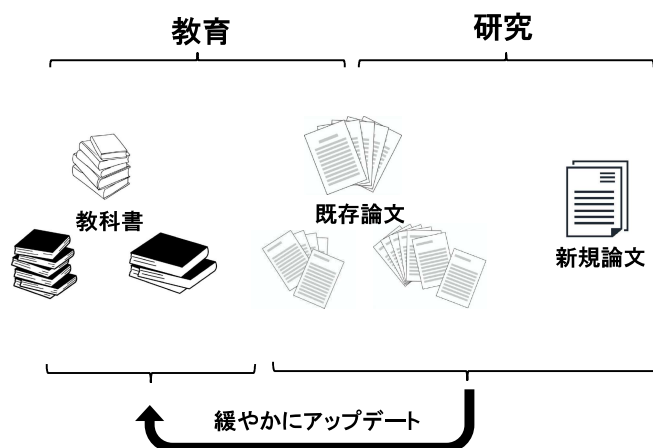


図 3: 従来の研究と教育の連携

ムーズになるし、最新の研究に沿った教育コンテンツを作り易くなる。そのような教育コンテンツを使った教育を受けた人材が研究を行うことで、そのフレームに沿った研究がなされ、それでまた教育へ転移できるコンテンツがオープンサイエンスの動きにより促進される。

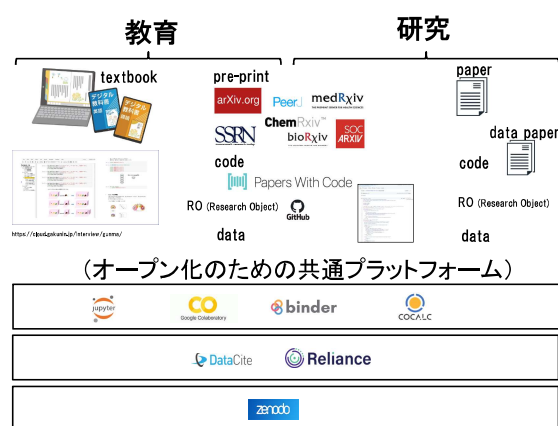


図 4: 教育と研究のプラットフォームの共通化例

このサイクルを実現することで図5に示す様に①新しい研究成果の教育への速やかな活用を進めることが可能となる。また、逆に②教育で活用したものを研究環境でも活用する流れも可能となり、教育で培われたリテラシーが直接的に研究の場で活用できる。

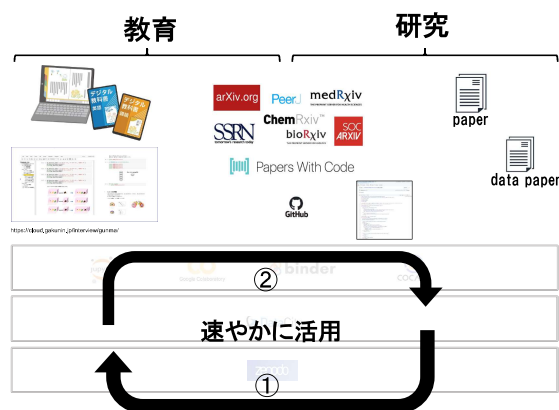


図 5: 教育と研究のコンテンツ流用サイクル

## 5 オープン化に関する教育と研究の連携例

前章で述べた枠組みにもとづいて実施した実験数学に関する教育から研究までの連携例について紹介する．本事例は素数分布に関連するもので，概要を図6に示す．

研究レベルの実験数学として，大域体における素イデアルの偏りについての現象を深リーマン予想との関係で説明した論文“Chebyshev’s Bias against Splitting and Principal Primes in Global Fields” [21] の Appendix に記載されている数値実験の再現と，Appendix に記載されていないけれど論文本文中で述べられている偏りに関する数値実験を実施した．この際利用したプログラムは [22] に公開されている．

教育レベルの実験数学としては，背景で述べた初等整数論のコンテンツで実行できる実験数学を用いている．また，それらの共通プラットフォームとしては実験実施環境としてデータ活用社会創成プラットフォーム mdx [23] を使用し，その各ノード上にコンテンツ技術を用いて構築した Jupyter Notebook 実行環境を利用している．

以下に実践内容について説明する．

### 5.1 実験準備

データ活用社会創成プラットフォーム mdx 上に 5 コンピュートノード (152 core, 243.44GB mem/node) からなる実験環境を図7のとおり構築した．各コンピュートノードには様々な大域体における偏りを計算するためのプログラムを配置し，必要に応じて必要なノードやコアを使って並列に実行できる構成とした．さらに，各コンピュートノードでの計算に共通に必要な素数表などのデータはオブジェクトストレージ上に配置し，偏り計算の結果の保存にもオブジェクトストレージを活用した．

素数分布に関する実験を数多く実施するので，実験準備の一つとして比較的大きな素数表 (10 の 14 乗まで) を生成し実験時に参照するために mdx 内にある S3 準拠のオブジェクトストレージに格納した．10 の 8 乗までの素数をそれぞれ 8bytes unsigned integer で表現した表を [https://s3ds.mdx.jp/prime\\_tables/prime\\_0](https://s3ds.mdx.jp/prime_tables/prime_0) という名前のオ

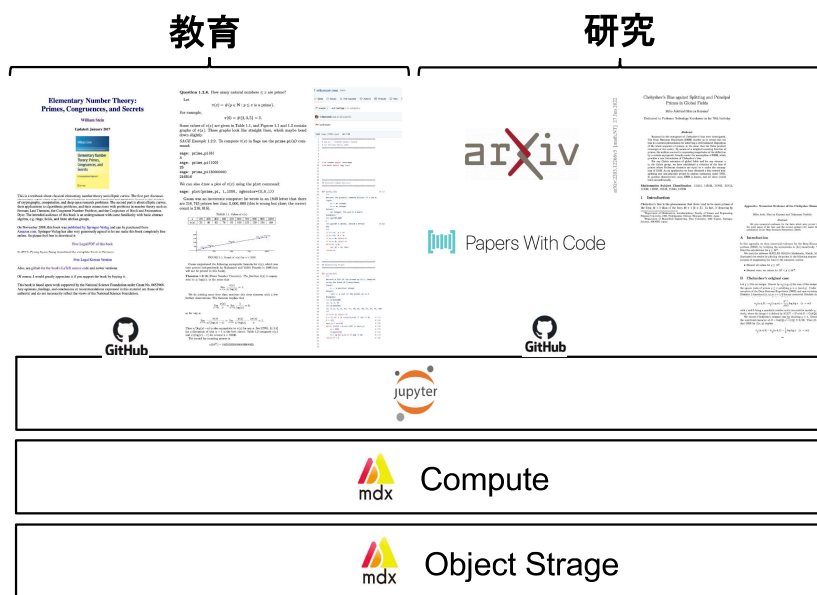


図 6: 素数分布に関する教育と研究

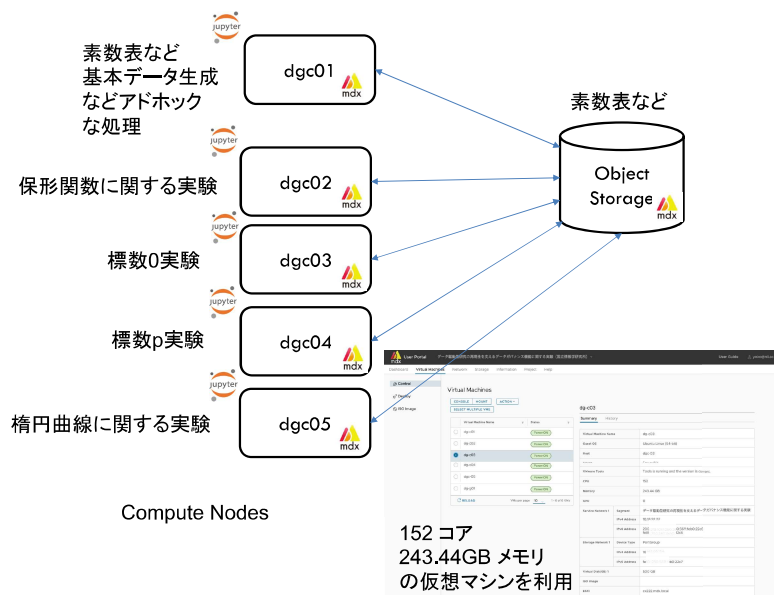


図 7: mdx 上の実験環境

プロジェクトとして格納し、それ以降の素数も 10 の 8 乗幅に同等に分割して、それぞれ順に [https://s3ds.mdx.jp/prime\\_tables/prime\\_1](https://s3ds.mdx.jp/prime_tables/prime_1) などという名前のオブジェクトとして格納した。全体で 10 の 6 乗個のオブジェクトに分割して、20TB 程度の素数表を作成した。これらのオブジェクトは上記 URL で公開されているので、これ自体もオープンデータとして他の教育や研究での活用が可能である。

また、論文の中で扱われている各種の大域体における素イデアルの偏りに関する実験を実行ノードを分けて並行して行えるようにした。ここではそれら実験の中で標数 0 の

例の一つである有理素数の虚 2 次体拡大での素イデアル分解における偏りに関する現象を確認するための実験について説明する. その他の実験については [22] を参照願いたい.

## 5.2 実験実施

実験は, “実験数学を Jupyter Notebook でもっとやってみる [13]” で説明したようなパラメータ管理と並列実行管理の仕組みを利用し実施した. パラメータ管理表の例を図 8 に示す. 図 8 の B 欄は虚 2 次体の判別式を示し, C 欄は虚 2 次体の類数を示す. この類数が偏りの度合いに関係する. F 欄は計算する範囲を示す. G 欄は並列実行する際に使用する最大コア数を管理する. 実行ステータスは G 欄で管理し, 例えば “d” は実行完了 (done) を意味する. パラメータ管理と並列実行管理詳細は上記先行研究 [13] を参照願いたい.

	A	B	C	D	E	F	G	H	T
1	-5	-20	2	0	$2 \cdot 10^{10}$	100	d	2022-06-06 11:30:26.518615+09:00	
2	-6	-24	2	0	$2 \cdot 10^{10}$	100	d	2022-06-06 13:29:17.332044+09:00	
3	-5	-20	2	0	$2 \cdot 10^{10}$	100	d	2022-06-06 15:24:01.553329+09:00	
4	-6	-24	2	0	$2 \cdot 10^{10}$	100	d	2022-06-06 17:16:49.910974+09:00	
5	-5	-20	2	0	$2 \cdot 10^{10}$	100	d	2022-06-06 19:11:25.327065+09:00	
6	-6	-24	2	0	$2 \cdot 10^{10}$	100	d	2022-06-06 21:04:00.727327+09:00	
7	-10	-40	2	0	$2 \cdot 10^{10}$	100	d	2022-06-06 22:58:32.001296+09:00	
8	-13	-52	2	0	$2 \cdot 10^{10}$	100	d	2022-06-07 00:52:12.494182+09:00	
9	-14	-56	2	0	$4 \cdot 10^{10}$	100	d	2022-06-07 02:43:32.130014+09:00	
10	-15	-15	2	0	$2 \cdot 10^{10}$	100	d	2022-06-07 04:35:23.403063+09:00	
11	-17	-68	2	0	$4 \cdot 10^{10}$	100	d	2022-06-07 06:26:18.767021+09:00	
12	-21	-84	3	0	$4 \cdot 10^{10}$	100	d	2022-06-07 08:17:20.111309+09:00	
13	-105	-420	4	0	$8 \cdot 10^{10}$	100	d	2022-06-07 10:06:41.710074+09:00	
14	-1155	-1155	4	0	$8 \cdot 10^{10}$	100	d	2022-06-07 11:54:05.033625+09:00	
15	-15015	-15015	5	0	$96 \cdot 10^{10}$	100	d	2022-06-07 13:45:42.709301+09:00	
16	-255255	-255255	6	0	$256 \cdot 10^{10}$	100	d	2022-06-07 15:32:58.190766+09:00	
17	-4849845	-19399380	6	0	$1536 \cdot 10^{10}$	100	d	2022-06-07 17:24:04.063192+09:00	
18	-111546435	-111546435	8	0	$2688 \cdot 10^{10}$	100	d	2022-06-07 19:14:45.928629+09:00	
19	-3234846615	-3234846615	9	0	$39424 \cdot 10^{10}$	100	d	2022-06-07 21:08:40.423953+09:00	
20	-22	-88	2	0	$2 \cdot 10^{10}$	100	d	2022-06-07 23:59:34.340797+09:00	
21	-23	-23	1	0	$3 \cdot 10^{10}$	100	d	2022-06-08 00:52:18.812187+09:00	
22	-26	-104	2	0	$6 \cdot 10^{10}$	100	d	2022-06-08 02:41:01.893458+09:00	
23	-29	-116	2	0	$6 \cdot 10^{10}$	100	d	2022-06-08 04:31:58.512194+09:00	
24	-30	-120	3	0	$4 \cdot 10^{10}$	100	d	2022-06-08 06:22:25.020917+09:00	
25	-31	-31	1	0	$3 \cdot 10^{10}$	100	d	2022-06-08 08:13:30.556852+09:00	
26	-105	-420	4	0	$8 \cdot 10^{11}$	100	n		
27	-1155	-1155	4	0	$8 \cdot 10^{11}$	100	n		
28	-15015	-15015	5	0	$96 \cdot 10^{11}$	100	n		
29	-255255	-255255	6	0	$256 \cdot 10^{11}$	100	r		
30	-15015	-15015	5	0	$96 \cdot 10^{11}$	100	w		

図 8: 虚 2 次体の実験のために利用したパラメータと並列実行管理表

各パラメータ毎に図 9 に示す Jupyter Notebook が呼ばれ計算実行する. 今回は Sage で記述したプログラムが実行され, 指定した範囲の計算が完了後計算結果をグラフ化し, その情報をオブジェクトストレージに書き出す.

## 5.3 実験結果

いくつかの判別式に対する計算結果について図 10 に示す. 論文で深リーマン予想を仮定した際に想定される素イデアル分解の偏りと計算結果はこの範囲では確認できている.

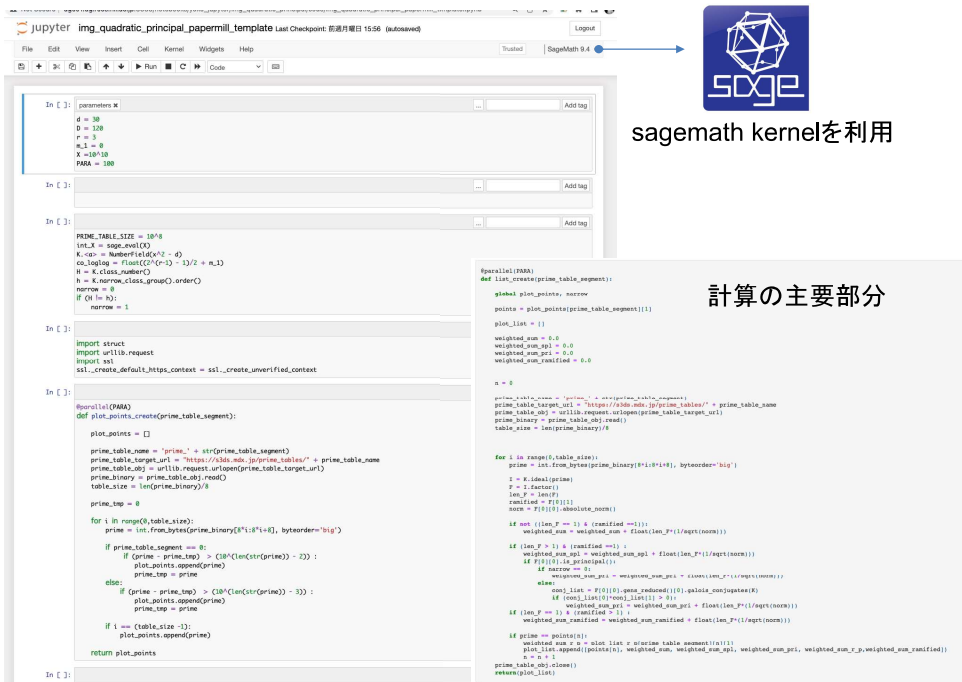


図 9: 虚 2 次体の実験のための Jupyter Notebook

## 6 考察

今回の連携例では、教育側のコンテンツは既存のものを想定して、その連携先としての研究側の実験を構成することに注力した。このため、一定の連携性を実現できたもののさらにスムーズな連携を実現するためには教育側のコンテンツについて研究側での知見をもとに再構築することが考えられる。具体的にはマルチコアを前提とした並列計算の仕組みを活用することやパラメータ管理と並列実行管理の仕組みの適用などである。

さらに、教育側のコンテンツをさらに初等的なものから出発する方向も考えられる。ハイブリッド実験数学をやってみる [24] で説明したように、身体的な実験数学から始めて、最新研究までをオープンな世界から連携させることもあり得るであろう。

## 7 謝辞

論文 “Chebyshev’s Bias against Splitting and Principal Primes in Global Fields” の Appendix に記載されている数値実験の再現と、Appendix に記載されていないけれど論文本文中で述べられている数値実験を実施にあたって、この論文の著者の一人である島根大学 総合理工学部 数理科学科 青木 美穂 教授に常に有益な助言をいただけたことに感謝いたします。このご協力がなければ、5章で説明した実験数学は実現できませんでした。



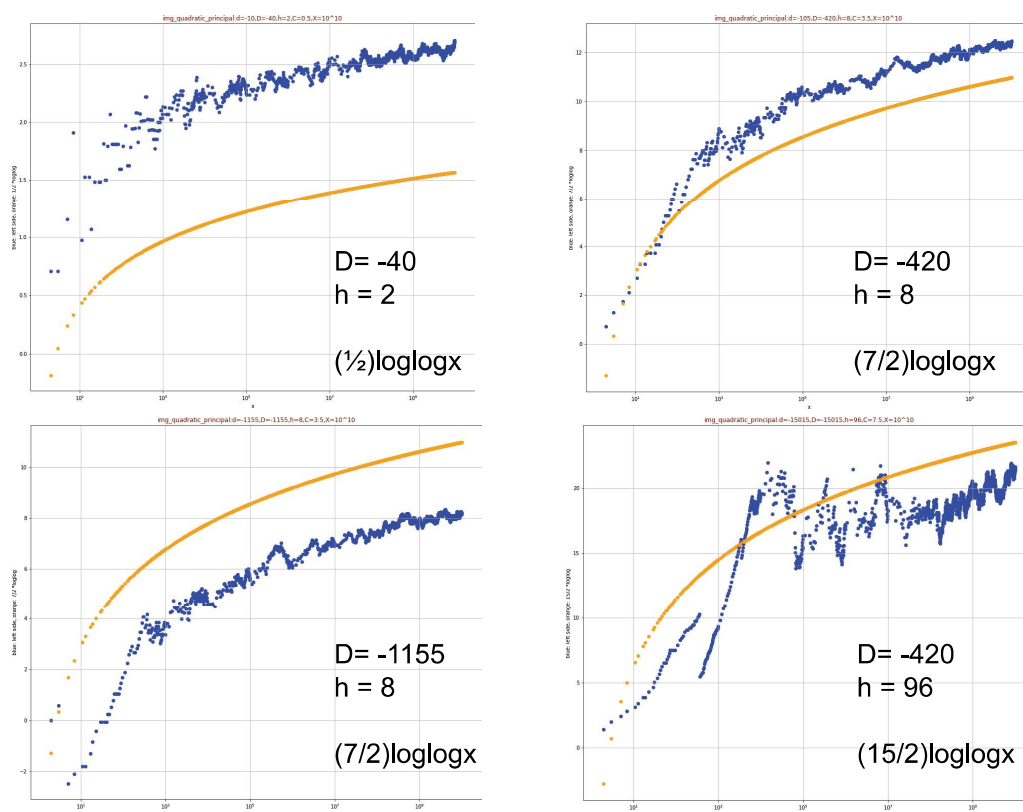


図 10: 実験結果の一部

## 参考文献

- [1] 山本芳彦, 実験数学入門, 岩波書店, 2000.
- [2] 清水克彦, 実験数学による創造性の育成についての検討: テクノロジーによる帰納・類比, そして推測の導入, 日本科学教育学会年会論文集 / 34 巻 (2010).
- [3] 小池正夫, 実験・発見・数学体験, 数学書房, 2011 年.
- [4] 清水克彦, 目代充寿, Mathematica for Student を用いた実験数学の教材開発, 数理解析研究所講究録 第 1780 巻 2012 年 213-220.
- [5] 松本 昌也, 清水 克彦, Google Colaboratory を用いた実験数学教材の開発: Python で完全数・メルセンヌ素数を探究する, 数理解析研究所講究録 2208, 20-30, 2021.
- [6] CoursewareHub, <https://github.com/NII-cloud-operation/>, (参照 2022-11-11).
- [7] 長久勝, 政谷好伸, 合田憲人. Notebook による講義・演習環境の開発. 第 27 回教育学習支援情報システム研究会 2019 年 3 月 22 日 情報処理学会.
- [8] 横山重俊, 浜元信州, 政谷好伸, Jupyter Notebook を活用したアクティブラーニングへのトライアル 暗号技術教育を例に, 2019 年度 数学教育学会 夏季研究会 (関西エリア).

- [9] 横山重俊, 浜元信州, 政谷好伸, Jupyter Notebook を活用した実験数学におけるリアルタイム進捗収集ツール, 2019 年度 数学教育学会 夏季研究会 (関東エリア) .
- [10] 横山重俊, 浜元信州, 政谷好伸, 合田憲人, Jupyter Notebook を活用した情報教育実践, 情報処理学会 情報教育シンポジウム SSS2019.
- [11] 横山重俊, 浜元信州, 政谷好伸, Jupyter Notebook を活用した実験数学環境に関する提案, 2019 年度 数学教育学会 数学教育学会 秋季例会, 2019.
- [12] 横山重俊, 浜元信州, 長久勝, 政谷好伸, 合田憲人, 実験数学を Jupyter Notebook でやってみる, 数理解析研究所講究録 2142, 79-91, 2019.
- [13] 横山, 重俊, 浜元, 信州, 長久, 勝, 谷沢, 智史, 藤原, 一毅, 政谷, 好伸, 竹房, あつ子, 合田, 憲人, 実験数学を Jupyter Notebook でもっとやってみる, 数理解析研究所講究録 2178, 48-57, 2021.
- [14] 中村憲, 整数論のソフトウェアとデータベースについての提案, 数理解析研究所講究録 759, 118-124, 1991.
- [15] 横山俊一, 数論データベース LMFDB の開発について, 代数学と計算, 2015.
- [16] Anderson, J. and Keahey, K.: A Case for Integrating Experimental Containers with Notebooks, 11th IEEE International Conference on Cloud Computing (Cloud- Com 2020).
- [17] Samit Dasgupta, Mahesh Kakde, Brumer-Stark Units and Hilbert’s 12th Problem, arXiv:2103.02516, 2021.
- [18] Papers With Code, <https://paperswithcode.com/>, (参照 2022-11-11).
- [19] William Stein, “Elementary Number Theory: Primes, Congruences, and Secrets”, <https://wstein.org/ent/>, (参照 2022-11-11).
- [20] Elementary Number Theory: Primes, Congruences, and Secrets, <https://github.com/williamstein/ent/>, (参照 2022-11-11).
- [21] Miho Aoki, Shin-ya Koyama, Chebyshev’s Bias against Splitting and Principal Primes in Global Fields, arXiv:2203.12266, 2022.
- [22] Chebyshev’s Bias Numerical Experimentations, <https://github.com/jxta/chebyshevs-bias/>, (参照 2022-11-11).
- [23] 大学・研究機関で共創する産学官連携のためのデータプラットフォーム mdx, <https://mdx.jp/>, (参照 2022-11-11).
- [24] 横山重俊, ハイブリッド実験数学をやってみる, 数理解析研究所講究録 2208, 10-19, 2021.