実験数学を Jupyter Notebook でもっとやってみる

群馬大学・総合情報メディアセンター　横山 重俊
Shigetoshi Yokoyama, Library and Information Technology Center, Gunma University
群馬大学・総合情報メディアセンター　浜元 信州
Nobukuni Hamamoto, Library and Information Technology Center, Gunma University
ライフマティックス株式会社　長久 勝
Masaru Nagaku, Lifematics
株式会社 ボイスリサーチ　谷沢 智史
Satoshi Yazawa, Voice Research
国立情報学研究所 藤原 一毅
Ikki Fujiwara, National Institute of Informatics
国立情報学研究所　政谷 好伸
Yoshinobu Masatani, National Institute of Informatics
国立情報学研究所 竹房 あつ子
Atsuko Takefusa, National Institute of Informatics
国立情報学研究所　合田 憲人
Kento Aida, National Institute of Informatics

1　はじめに

群馬大学ではアクティブラーニングの取り組みの一環として，実験数学的手法を用いた教材を開発し，受講生主体の実験数学を授業に取り入れている．実験数学 [1] の計算機環境として JupyterHub[2] ベースの CoursewareHub[3, 4] と呼ばれる Jupyter Notebook 実行基盤を活用して教育実践を行なって来た [5, 6, 7, 8, 9].

本報告では，実験数学環境をさらに工夫することで，実験精度のコントロールや各種パラメータ変更の管理ができ，さらに複数ノードで分散処理できる環境を提供し，先行研究「実験数学を Jupyter Notebook でやってみる」[10] の事例より高度な実験数学を実現するための方式提案を行う．

ここで高度な実験数学というのは，先行研究で実践例として述べたフェルマの小定理に関する実験のように，証明が知られている古典的な定理の検証的な実験ではなく，“予想”と呼ばれることが多い，より研究に寄った領域に属する実験を指す．

提案するのは，Jupyter Notebook の実行環境である BinderHub[11] と Jupyter Notebook のコマンドライン起動ツールの Papermill[12] を組み合わせることで，実験精度のコントロールや各種パラメータ変更の管理ができ，複数ノードで分散処理できる環境をクラウドサービス上に簡便に構築できる実装方式である．また，この実験数学環境上で BSD 予想 [13] に関する実験を実施したので，その実験模様を実践例として紹介する．

2　背景

先行研究では，群馬大学で実施されている講義「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アクティブラーニングの取り組み内容について報告した．特に，我々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，具体例で報告した．本報告では，前章で述べたように，より研究に近い高度な実験数学，すなわち「もっとやってみる」の領域を対象とする．

「もっとやってみる」の領域を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来ている中，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．先端的研究に近い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．我々は，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．

3　課題

先行研究で利用した Jupyter Notebook 実行基盤 CoursewareHub では，実験環境として一つのコンテナ内に収容できる実験が簡便な手順で実行でき教育に活用できた．しかも，実験環境をコンテナイメージとして流通させ，各所の CoursewareHub で利用することができるため，実験の再現性確保も比較的容易であった．

一方，今回のターゲットである高度な実験数学の場合，その実験環境は一般的に計算量が多い，実験パラメータが多い，実験環境が多様である，などの特徴があるため，先行研究の制約である一つのコンテナイメージ内に実験数学環境を収容しなければいけないといった枠を超えた取り組みが必要となる．このため研究レベルの実験環境を教育に適用することに取り組むことが課題となる．

この際，研究者が環境の利用者であった場合に加えて，さらに解決しなければならない課題が存在する．一つは，教育利用を目指すためには，高度な実験実行まで道のりが長くなり，さらにその道のりの中で超えなければならない高くなりがちなハードルを下げることである（簡便性）．もう一つは，教育時には時には多数の受講生により一斉に実験が実施されることに配慮しなければならないことである（バースト耐性）．

4　アプローチ

「背景」で触れたように，本報告では研究分野をまたがったオープンサイエンスの流れに沿った課題解決へのアプローチを取る．

4.1　データ中心型研究の再現方法

データ中心型研究の研究プロセスを単純化すると，(1) データの蓄積, (2) 実験（データ分析と発見）, (3) 論文化となり，(1) と (3) についてはオープン化が進んでいる．残る (2) についてのオープン化を進める手段が求められている．

研究データの再利用を進めるためにも，この研究データから論文に公表したデータ分析結果にたどり着くまでの実験プロセスを公開し共有する必要がある．そのデータ分析プロセスの中には，実験手順だけではなく，その実験手順に従ってデータ分析を行える実験環境をどう構築するかについて記述した手順も必要である．

これらの手順を保存・管理・提供する情報基盤を新たに構築することで再構成に必要であった手間を大幅に削減できる可能性がある．すなわち，実験手順と実験環境を共有することで，派生研究者が論文に書かれている情報からオリジナル研究者の用いた実験手順と実験環境を手動で再構成するという，手間のかかる作業から解放されるのである．

この際, 論文，研究データ，実験手順は何らかのストレージサービスに長期保存し，さらにそこに恒久的な識別子などを付与することが比較的容易である．しかしながら，実験環境（E：Environment）はハードウェアとソフトウェアの総体であり，その上で実験が実験手順に従って研究データにアクセスしながら実行できる必要がある．これは単純に長期保存できるものではないので，それを長期保存後再構築する手段が必要になる．

実験を共有・流通させるための仕組みとして，まず実験を「各分野で標準化された実験環境テンプレート（以後 実験環境テンプレート, ET: experiment Environment Template）」を使って行う実験環境の構築とその実験環境の上で「各実験を実行するための再現情報（以後 実験ノート, N: experiment Note）」を使って行う実験実施より構成する．図１に示すように，論文と同時に各分野においては標準化された実験環境テンプレートが共有化され，実験ノートを共有・流通することで，実験の再現性が確保される可能性を追求する流れに沿って検討を進める．

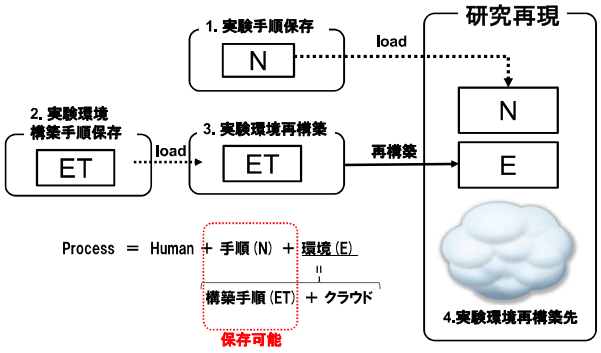


図 1: 実験再現情報の保存と再現方法

4.2　教育への展開

この枠組みを研究においてのみ適用するのであれば，多くの関連研究があり，様々な実装方式が提案されている [17]．但し，教育への展開に不可欠な簡便性とバースト耐性の確保という観点から，これらの既存の実装方式を評価した場合，特に簡便性の観点で最適なものを見つけるとすると BinderHub を活用した方式しか採用できない．

BinderHub は利用者が実験に必要な Jupyter Notebook とそれを実行できる環境を構築するための情報の所在を伝えるだけで，自動的にその構築情報（ET）を元にコンテナイメージを build し，JupyterHub 上にコンテナを起動後，利用者が指定した Jupyter Notebook（N）が利用できる状態にする．実験のためには上記情報の所在を表す URL を共有し，その URL を BinderHub サービスに渡すだけで良い．例えば各授業の LMS コース内の該当箇所に URL を配置するだけで簡便に実験数学へナビゲーションできる．

さらに BinderHub サービスは mybinder.org などクラウドサービスとして無償提供されていて容易に利用することができるし，バックエンドの実行基盤である JupyterHub は各所のクラウドサービス内に分散配置されており，さらにそれぞれ十分なリソースを持っていると考えられるため，バースト耐性も保持しているものが必要に応じて選択できる．

但し，この方式では実験環境が一つのコンテナイメージに収まるような先行研究で扱ったレベルの実験数学を提供することは可能であるけれど，本報告の対象とする高度な実験数学を提供することは難しい．従って，この実装方式の持つ簡便性とバースト耐性という特徴を継承しつつ，高度な実験数学が実行できる実装方式を見出す必要がある．例えば，mybinder.org は認証なしで利用できて，無償で，大きなリソースを提供してくれているけれど，そういう運用を可能とするために各利用者の利用可能セッション時間は短いという制約も持つ．つまり，教育のシーンに依存して要求される要件が決まり，最適な BinderHub サービスは変化するので，場合によっては教育機関自らが自前の BinderHub サービスを構築することが必要になる可能性もある．

5　提案実装方式

計算量や実験パラメータの多さに対応するために分散処理の仕組みを導入する．そのためには，計算量を制御するための計算精度に関する情報や各種パラメータ情報を管理し，それらを集中管理しつつ，実際の計算は分散し並行実行できる必要がある．このため，これらの実験毎に存在する Jupyter Notebook 内の変化要素を各 Jupyter Notebook から分離し，Jupyter Notebook 外に間接化する． つまり Template Notebook とそれを制御する パラメータ群に分離して管理する構造を付加することとする．具体的には精度制御を含めたパラメータ群と各実行 Notebook の実行状況を表形式（例えば Ethercalc[18] のような表管理クラウドサービス）で管理し，それらを Papermill が持つ Jupyter Notebook 実行時のパラメータ制御機能を結びつける．

これら二つのツールを統合化することで，今回提案する，計算量が多い，実験パラメータが多い，実験環境が多様である，高度な実験数学を実施できる環境が既存インフ

ラを組み合わせることで図 2 に示すように実現できる．Papermill による並列計算の起動部分自身も Jupyter Notebook として記述し，これを BinderHub で実行することで実際の実験の Jupyter Notebook を並行実行することで統合することができる．また，各 Jupyter Notebook で得られる実験結果は外部のストレージに保存する．

6　実践

実験例として BSD 予想の基本的な部分である楕円曲線の rank とその楕円曲線の素数 p を法とした時の零点の数 N_p の素数 p を渡った分布の関係を調べる．具体的には，$\prod_{p<C}(\frac{N_p}{p})$ という積について実験精度である C を大きくして行った時の振る舞いを実験対象として選択し，実装および実験を行なった．

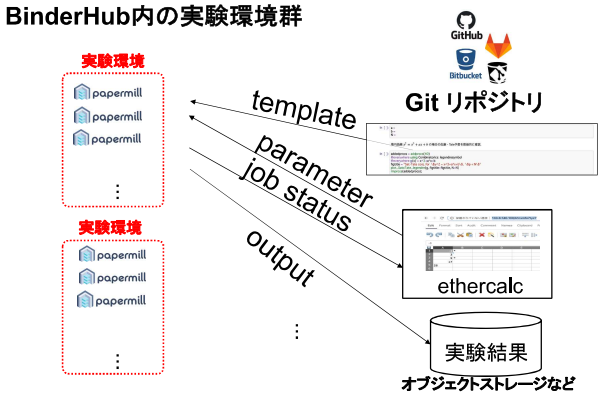


図 2: 提案実装方式

6.1　事前準備

予め rank の高い楕円曲線を探索して，効率的な実験数学とするようなパラメータ収集が必要である．具体的には楕円曲線群 $y^2 = x^3 + 8x + B$ の rank 計算を Jupyter の SageMath kernel を使って計算し rank 5 までの楕円曲線例を収集した．その収集結果から rank=0, 1, 2, 3, 4, 5 のそれぞれの楕円曲線のうちいくつかを rank 毎に適宜選択することとした．元々今回の実験例では記述言語として計算速度を考慮して Julia を選択しているけれど事前準備の際に探索した範囲では楕円曲線の rank 計算のための便利な関数を SageMath の中でしか発見できなかったために，このような実装とした．Jupyter Notebook の特徴として様々な kernel を従えることができるので，適宜実験フェーズに合わせた選択が可能な点も便利な特徴である．

6.2　実験例

実験例となる Jupyter Notebook 群とパラメータ管理表を図 3 に示す．この実験例の Jupyter Notebook 群は Julia kernel 上で動作する．この実験例では $y^2 = x^3 + Ax + B$ で表現される楕円曲線についての実験を繰り返す．係数の A，B についてはそれぞれパラメータ管理表の A 欄と B 欄を使って管理する．なお，今回の報告では $A = 8$ の場合について説明する．また，実験精度を表現する素数 p の取りうる範囲を決める C についてはパラメータ管理表の C 欄で管理する．各楕円曲線の rank については D 欄に記述しておく．さらに，分散処理している各実験の状態管理をするために S 欄を使った．w は wait 状態，r は run 状態，d は done 状態を表現している．

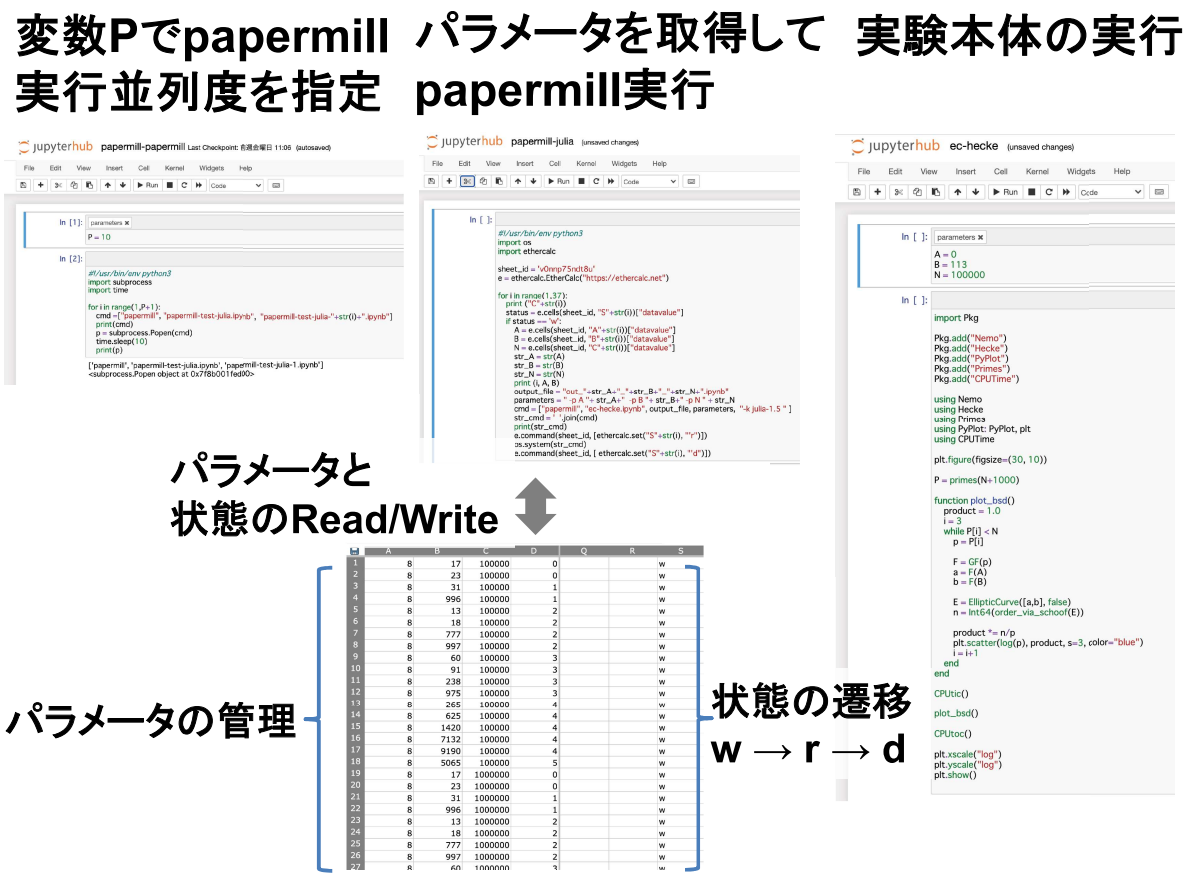


図 3: 実験例

なお，本実験例をこちらのページ https://github.com/jxta/bsd/blob/master/launch.md にある起動ボタンを押下するだけで BinderHub を呼び出し Papermill を起動し，当該実験数学を再現することができる．

6.3 実験結果例

実験結果例を図 4 に示す．計算例の各楕円曲線の rank と素数 p を法とする楕円曲線上の点の数の関係性が観察できる．Y 軸は $\prod_{p<C}(\frac{N_p}{p})$ を表し，X 軸は C を表す．X-軸は $log(log(X))$ スケールに，Y-軸は $log(Y)$ スケールにして $\prod_{p<C}(\frac{N_p}{p})$ の傾向を把握し易くしている．BSD 予想はデータは傾きが曲線の rank に等しい直線をなすことになる．

これらの結果のうち $rank = 3$ の例である $y^2 = x^3 + 8x + 60$ の $C = 100000$ までの実験結果は，図 5-左に示すように，予想に沿ったものであることが分かり易い．しかし，$rank = 5$ の例である $y^2 = x^3 + 8x + 5065$ の $C = 100000$ までの実験結果は図 5-右のようになり，この結果の傾きが $rank = 5$ に沿ったものであるのかはすぐには判然としないかもしれない．これは $C = 100000$ 程度の実験しか実施していないことが原因である

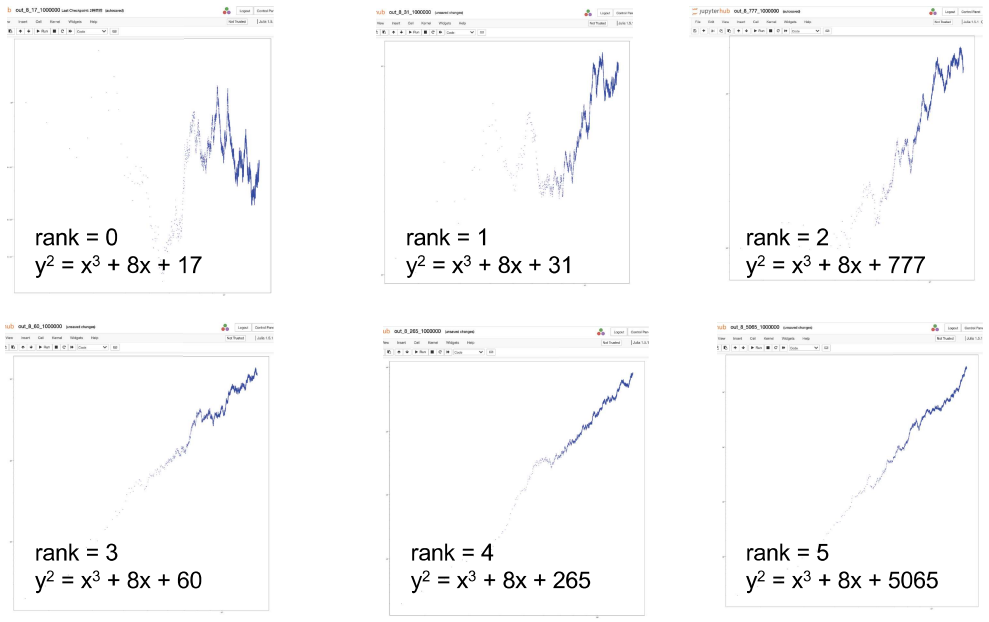


図 4: 実験結果例

可能性もあり，さらに大きな素数までのデータを収集してみたくなる実験結果である．

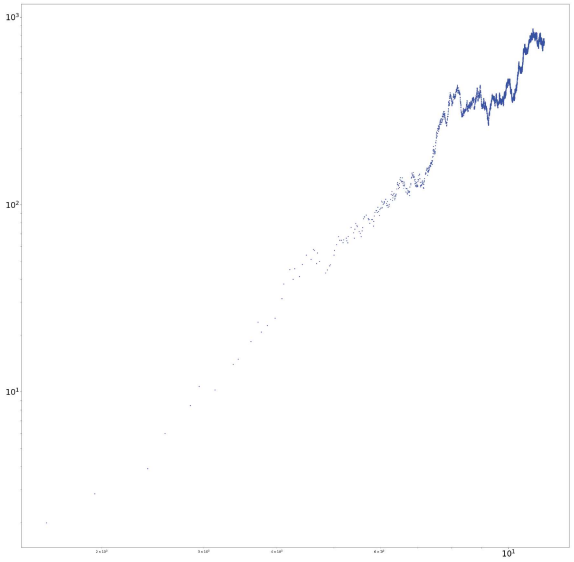
そのためアドホックにパラメータ管理表を変更して再度実験を実行し，$C = 1000000$ までの実験を実施してみた．その結果は図６のようになり，より予想を裏付ける結果に近付いたと実感できたりするのである。このようなインタラクティブでイテレーティブな実験数学が簡便に実施できるのは，パラメータ管理表を共同編集容易なクラウドサービスにした実装のメリットである．

7　評価

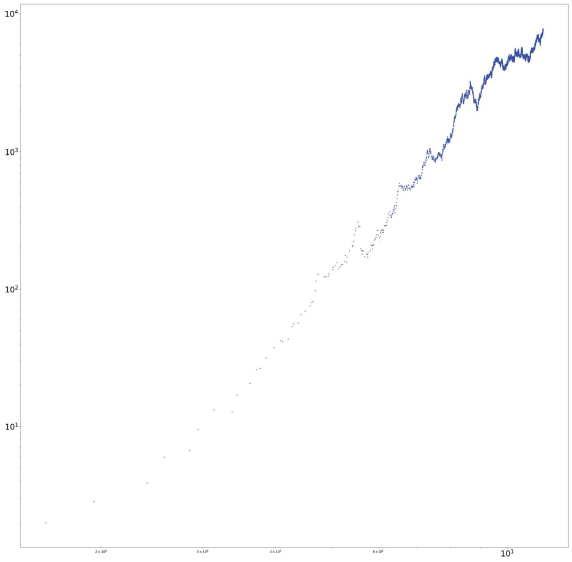
今回紹介した BSD 予想に関係する実験の他，佐藤予想 [19] や深リーマン予想 [20] に関する実験についても実践している．これらの数論における類似した実験を実践した範囲では，実験の再現性の確保やその実施の容易性などは確認できている．但し，これらの実験は実験分野の幅としては狭く，さらに広い範囲の実験について同様に効果があるかどうかについては実践例を増やして確認していく必要がある．

また，教育実践については未着手であり，今後機会を得て実施し，教育的視点からの評価が必要になる．

さらには，教育への適用という制約を付けたための方式選択が，研究という視点からは大きな制約となり，実用的でないという懸念もある．そのため研究視点での実践も実施し，両視点からのすり合わせを継続的に行っていく必要がある．例えば，研究視点では，いわゆるスーパーコンピューターなどで利用されるジョブキューイングシステムの利用や GPGPU の活用などが今後さらに進んでいくと考えられるけれど，これらと教育利用で要求される簡便性とバースト耐性を両立させることができるか，については未着手な課題である．



$y^2 = x^3 + 8x + 60$, $C = 100000$
$(rank = 3)$



$y^2 = x^3 + 8x + 5065$, $C = 100000$
$(rank = 5)$

図 5: 実験結果例（考察）

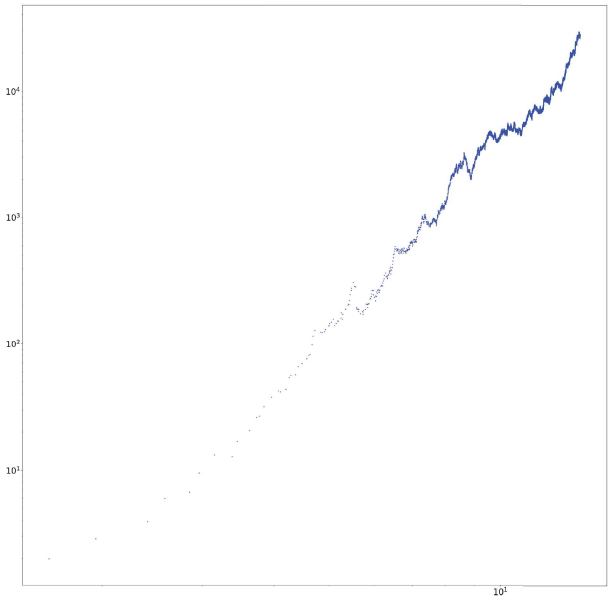


図 6: $y^2 = x^3 + 8x + 5065$, $C = 1000000(rank = 5)$

8　今後の進め方

オープンデータ管理の基盤として現在国立情報学研究所で開発が続いている Research Data Cloud の成果を教育へ活用するという取り組みに，本報告で述べた活動および「評価」の最後で述べた研究視点と教育視点のすり合わせによる成果をつなげて行きたいと考えている．

その際，近年普及が期待されているオープンデータを含む研究データが蓄積されている学術クラウドとの連携も重要である．例えば，今回の実験例では各 Jupyter Notebook 内にオンデマンドで生成していた素数表のような大きな普遍的データの共有とそれらをを使った実験数学の蓄積を進めたい．

参考文献

[1] 山本芳彦, 実験数学入門, 岩波書店, 2000.

[2] JupyterHub, https://jupyter.org/hub, (参照 2021-02-01).

[3] CoursewareHub, https://github.com/NII-cloud-operation/ , (参照 2021-02-01).

[4] 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.

[5] 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）.

[6] 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）.

[7] 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.

[8] 桑田喜隆，石坂徹，小川 祐紀雄，政谷好伸，長久勝，横山重俊，浜元信州，第 24 回人工知能学会　知識流通ネットワーク研究会，2019 年 03 月.

[9] 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.

[10] 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.

[11] BinderHub, https://binderhub.readthedocs.io/, (参照 2021-02-01).

[12] Papermill, https://netflixtechblog.com/scheduling-notebooks-348e6c14cfd6, (参照 2021-02-01).

[13] Birch and Swinnerton-Dyer conjecture : https://en.wikipedia.org/wiki/Birch_and_Swinnerton-Dyer_conjecture, (参照 2021-02-01).

[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] 横山 重俊 , 浜元 信州 , 長久 勝 , 藤原 一毅 , 政谷 好伸 , 竹房 あつ子 , 合田 憲人, データ分析プロセス共有による研究再現例, 研究報告インターネットと運用技術（IOT）, IOT-51(8),1-7, 2020.

[18] Ethercalc, https://ethercalc.net/, (参照 2021-02-01).

[19] 佐藤幹夫, 私の数学 (数論へ-ラマヌジャン予想)，佐藤幹夫の数学, 日本評論社, 46-48 (2007).

[20] Kimura, T., Koyama, S. and Kurokawa, N.: Euler Products Beyond the Boundary, Letters in Mathematical Physics, Vol. 104, No. 1, 1-19 (2014).