Computational analysis of the density of integer multiplications and prime numbers in the range of $I = [1; 10^6]$

Joan Alcaide-Núñez

Mathematics Department, Deutsche Schule Barcelona, Av. Jacint Esteva i Fontanet, 105 08950 Esplugues de Llobregat Barcelona, Spain
joan.alcaide@dsbarcelona.com

October 24, 2024

Abstract

Prime numbers are a fundamental part of number theory in mathematics and have various applications. This paper aims to search for mathematical relations between the rates of prime numbers and the distribution of multiplication through the positive integers up to 10^6. We compute the distribution of prime numbers and multiplication combinations using Python scripts. Then we try to prove a couple of tests to show (anti-) proportional relation between the best-fit models (linear and exponential) of the data. We find it hard to show a correlation between the rate of prime numbers in stacks of $10,000$ and the near-constant rate of multiplication combinations. Also, we find a misalignment of the prime number theorem equation with our empirically computed data. We conclude that this lack of correlations is due to yet unknown patterns in prime numbers, not related to the distribution of multiplications.

1 Introduction

Prime numbers are a fascinating part of arithmetic. They have enabled many applications in physics (e.g. quantum mechanics), as well as computer science: random number generation [Den04], hash functions [AD13], and RSA encryption [RSA78] (the most common in the world). Mathematicians have been interested in prime numbers and their characteristics for centuries [Bri24]. In the 17th century, the infinitude of prime numbers was first proven by Euclid's theorem. The Prime Number theorem formalizes the distribution of primes through the positive integers. The equation expresses it

$$\pi(x) \sim \frac{x}{\log(x)}, \tag{1}$$

which was proven independently by Jacques Hadamard [Had96] and Charles Jean de la Vallée Poussin [DLVP96] in 1896 using Riemann's ζ function. The first values for the equation can be seen in table 1.

x	$\pi(x)$
10	4
10^2	25
10^3	168
10^4	$1,229$
10^5	$9,592$
10^6	$78,498$
10^7	$664,579$
10^8	$5,761,455$
10^9	$50,847,534$
10^{10}	$455,052,511$

Table 1: The first results from 10 to 10^10 for the prime number theorem (equation 1) in logarithmic scale.

In this paper, we take a numerical approach to the distribution of prime numbers for the first $1,000,000$ (10^6) positive integers. Note that we only use positive integers, as prime numbers by definition only can be positive integers. We aim to compute the distribution of prime numbers for this range and compare our results with the prime number theorem function. Moreover, we want to compute what we call the distribution of multiplication, that is the number of different combinations that a number can be calculated via the multiplication operation of positive integers. We search for a correlation for both distributions, as we expect the density of prime numbers to decrease over x and the quantity of multiplication combinations to increase.

We first describe our methods and present our code in section 2. Then we will show our results in section 3 and discuss them in section 4. Finally, we will wrap up with the conclusion in section 5.

2 Methods

To compute the distribution of multiplication and the distribution of prime numbers I have written a `Python` script, which is publicly available in GitHub [1]. The `Python` script is quite simple and an explanation of each step is embedded in `Jupyter Notebook`. In short, it iterates through $1,0000,000$ for i searching for prime numbers[2] and counting the all the different combinations of multiplication operations that can be done to achieve this value. Then it performs different tests, later described in this paper.

We cannot iterate for all infinity of numbers to check if they result in our value. It can easily be shown that to result in a product x, both factors f_1, f_2 of a multiplication $f_1 \times f_2 = p$ must be smaller if we define the three variables to be positive integers. Therefore, we only iterate through the positive integers smaller than x. Further, we can reduce the range to $\sqrt{x}$ as half of all possible divisors of x are to be found in this range (see equations below) [HW79]. We compute this loop and then add to the list f_1 and $f_2 = x/f_1$ to have the real count. Ultimately, it displays the data in the diagrams within this paper.

$$\text{the factors: } f_1, f_2 \in \mathcal{N}$$

$$\text{let a multiplication be } f_1 \times f_2 = x \quad ; x \in \mathcal{N}$$

$$\therefore f_1, f_2 < x$$

$$\text{also maximum factors are } f_1 = f_2 \implies f_{1,2} = \sqrt{x}$$

$$\therefore f_1, f_2 < \sqrt{x} \quad \square$$

3 Results

Although computationally lasting at first glance, this script can be run on an ordinary laptop (our case: M1 CPU and 16GB of memory) thanks to a couple of enhancements to the code for viability. The results were computed twice, one time for a range $[0; 1,000]$ and the second for the interval $[0; 1,000,000]$. We only present the results for the 10^6 iterations.

The results for the distribution of multiplications are shown in figure 1, in which we can see the mean in red. The best fit linear regression was also computed and follows the equation $y = 0.000003000339508304902 3x + 12.469816746017331$, here in purple.

In figure 2 we display the distribution of prime numbers in badges of $10,000$ (following left-handed y-scale) and the total quantity of prime numbers in turquoise (following right-handed y-scale).

Finally, in figure 3 we have plotted the prime number theorem function in green over our bar diagram, as well as an asymptotically best-fit model computed from the stacks in purple, which follows equation $y = -128390.42104270413 \times e^{-0.0000001587477443985961 7\times x} + 128479.10024659803$.

4 Discussion

In the following section, we are going to discuss the results presented in section 3. From figure 1 we can see that, though greater numbers have more divisors and there is a growing tendency, the reality is that the best fit has a slope of 3×10^{-6} which is negligible at large scales. Furthermore, we see that the median and the best-fit line

[1] distribution of multiplication Repository

[2] Using `sympy` library's `isprime()` method.

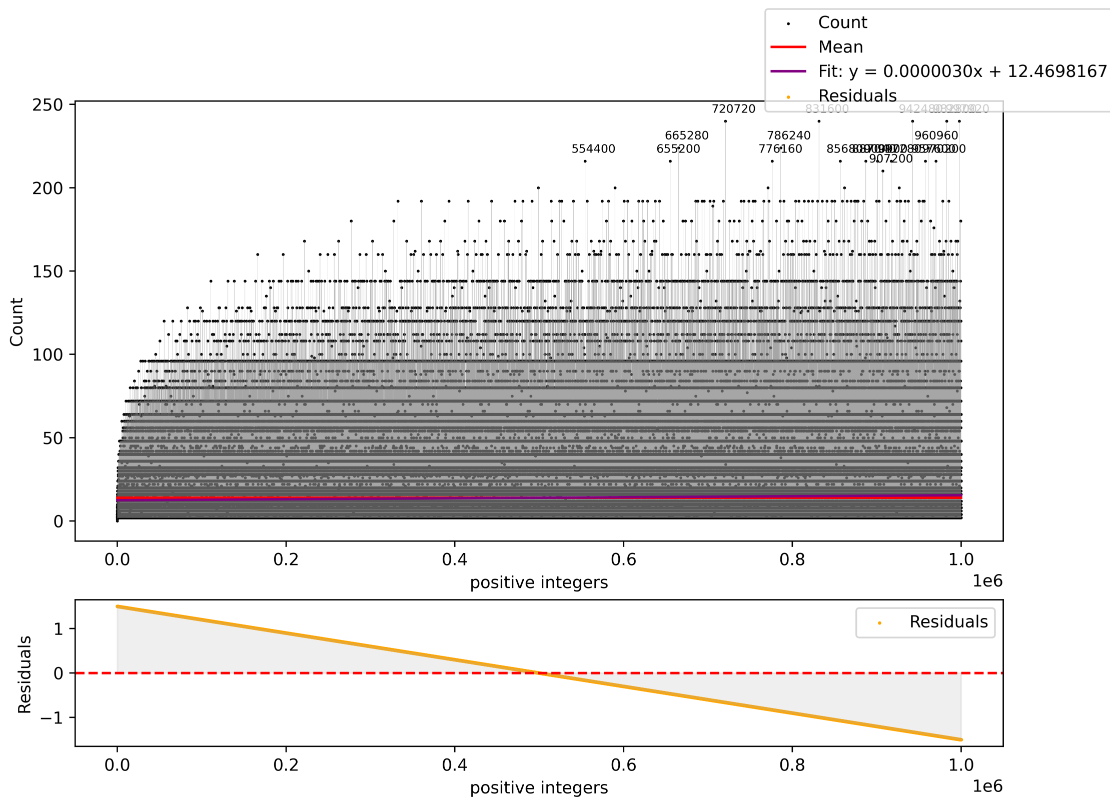

Figure 1: Multiplication distribution of the positive integers through 10^6. Numbers with more than 200 combinations are displayed alongside the data point. The mean is shown as the red line and the best fit linear regression following $y = 0.0000030003395083049023x + 12.469816746017331$ as the purple line. The residuals between the mean (red) and the best-fit linear model (purple) can be seen in the lower panel.

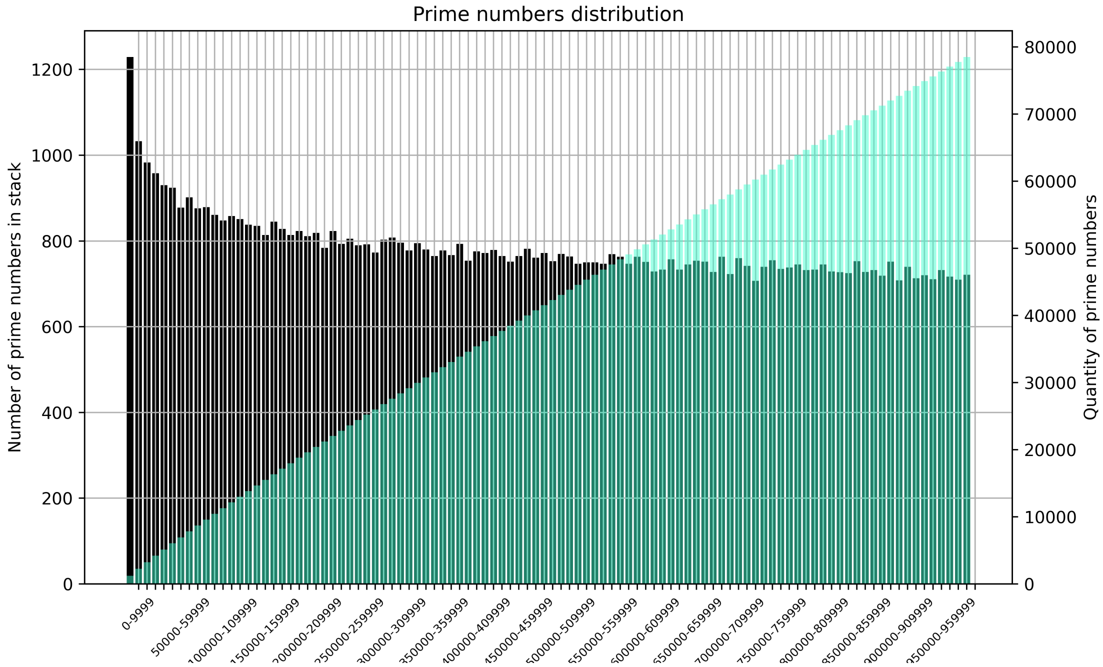

Figure 2: Distribution of prime numbers of the positive integers through 10^6. Bars represent the quantity of prime numbers in badges of $10,000$. In turquoise, the total number of prime numbers up to that stack.

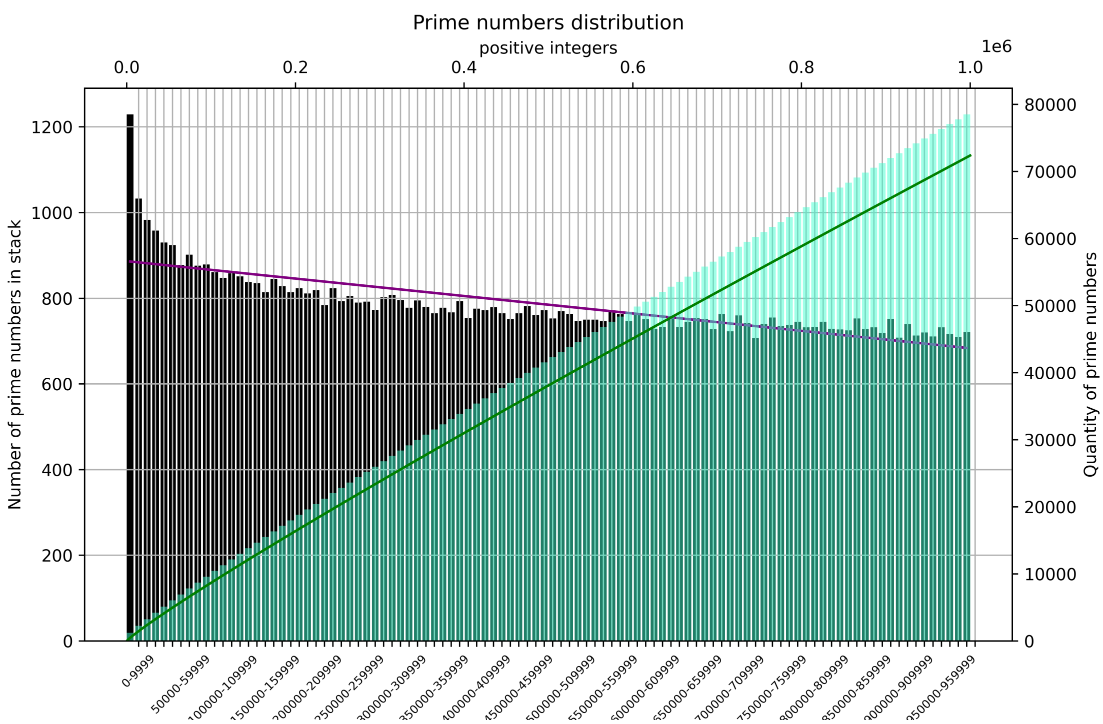

Figure 3: Same bar chart as figure 2, now also plotting the prime number theorem $\pi(x)$ equation 1 in green and the best fit asymptotic model for the quantity stacks in purple following $y = -128390.42104270413 \times e^{-0.0000001587477443985961 7\times x} + 128479.10024659803$.

compute in very small residuals, seen in the lower panel of figure 1. Therefore, we can affirm that the distribution of multiplication along the positive integers has a very low, near-constant increasing rate.

If we look at figure 2, we can see that the density of prime numbers decreases as the integers increase. We can confirm that from the rate slope of the best-fit model in figure **??**, which shows an exponentially decreasing function, asymptotically to ~ 1.5. We have used an exponential model to fit this data, as prime numbers are infinite. From the fact, that the function approaches 128479.10024659803 asymptotically (see appendix B A we show that prime numbers are infinite (at least taking into account $1,000,000$ iterations). However, when taking figure 3 into account, we see that the exponential decrease is nearly linear. Thus we can affirm that the distribution of prime numbers is exponentially decreasing through the first hundreds of thousand. Nevertheless, it stabilizes and becomes more linear around $300,000$ and forward, also recognizable in figure 2.

Finally, we can compare the turquoise bar chart with the prime number theorem function. We have found a correlation between both of them, as we take into account that $\pi(x)$ is an approximation. Although there is not a perfect match, the graph and the bars have similar rates. We attribute the small mismatch to the fact that we have computed only the first 10^6 integers, while $\pi(x)$ is an approximate function for the complete $]-\infty;\infty[$ range. Further, we have compute the mean squared deviation (MSD) following equation 2 to compute the prime theorem's error in reference to the total number of prime numbers. Where n is the upper limit, in our case 10^6, and Y_i the value for the respective functions.

$$msd = \frac{1}{n}\sum_{i=1}^{n}(Y_i - Y_{i'})^2 \tag{2}$$

We have computed the results to be $MSD = 14074333.039939027$.

We also want to search for a correlation between the decreasing rate of prime numbers (equation 3 and the 'increasing' rate of the distribution of multiplication (equation 4).

$$y = -128390.42104270413 \times e^{-0.0000001587477443985961 7\times x} + 128479.1002465980 \tag{3}$$

$$y = 0.0000030003395083049023x + 12.469816746017331 \tag{4}$$

We have plotted both lines in figure 4 in which we can see that the angle is near $90°$. If both functions were linear fits and were perpendicular to each other, it would mean that the decreasing rate of prime numbers would be anti-proportional to the increasing rate of multiplication combinations. Using equation 5 (see appendix C), we find the angle between both functions in their intersection to be $1.1684878226°$, and thus not proportional.

$$\theta = \arctan\frac{\mid m_1 - m_2 \mid}{1 + m_1 \times m_2} \tag{5}$$

A further test that we have performed is to compute the multiplication of the slopes of the graphs. If they were normal, the result should be $m_1 \times m_2 = -1$. However, we have computed a value of -0.0000000662, thus not normal.

$$\text{slope of exponential function: } m_1 = -0.0203937860310898$$

$$\text{slope of linear regression: } m_2 = 0.0000030003395083$$

$$m_1 \times m_2 = -0.0000000611882820$$

5 Conclusion

First, we have computed the distribution of prime numbers (empirically) and we have computed and defined the term distribution of multiplications through 10^6. We find it hard to relate the prime number theorem with our computations. We also have performed tests in the search for a relation between different best-fit functions to the distributions. We have not found any correlations either between the prime number theorem and the rate of prime numbers or the function of multiplication combinations and the rate of prime numbers. We attribute that lack of relation to yet unknown patterns in the distribution of prime numbers along the positive integers. Moreover, this data is to be assessed with caution, as we have computed values only for 10^6 iterations and we have computed best-fit models using `Python`. Also, take into account that this is a numerical-computational analysis.

Acknowledgements

I acknowledge the use of the `Python` libraries `NumPy` [HMVDW+20], `SciPy` [VGO+20], `SymPy` [MSP+17], and `Matplotlib` [HD07]. I would like to thank Susana de Mesa Sterhoff and Jürgen Goicoechea, PhD, for their feedback and commentaries. Also Ania Alvarez for her support and enthusiasm.

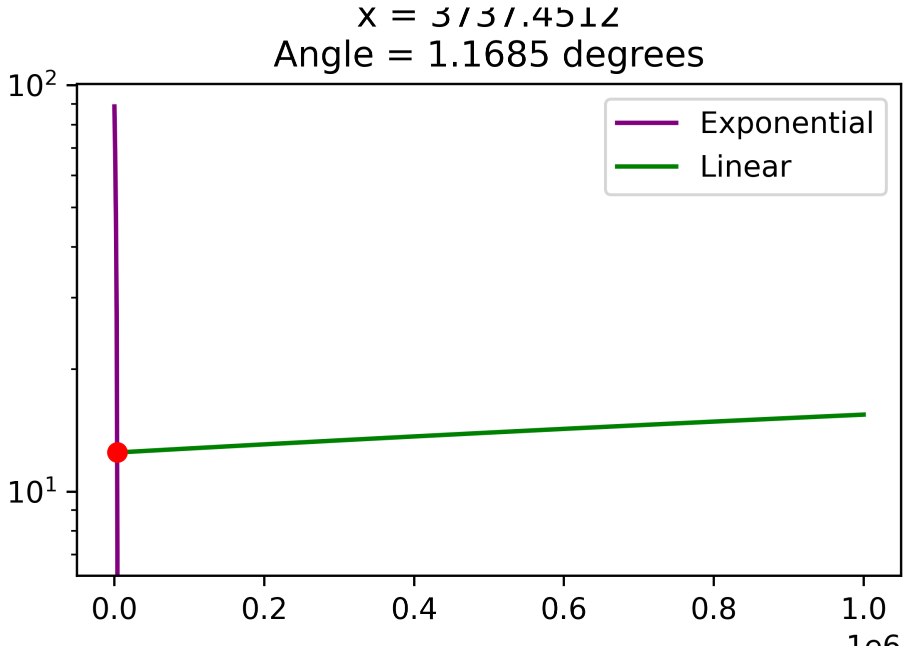

Figure 4: In this diagram we plot the best-fit model of the decreasing rate of prime numbers from figure 3 (equation 3) in purple and the increasing best-fit linear regression of multiplication combination from figure 1 (equation 4) in green. Then, we compute the value for the angle in the intersection of both functions.

A Appendix: Showing asymptotes of prime numbers density best-fit

In this appendix, we show that function 3 (the best-fit model to the 10^6 decreasing prime density data) is asymptotic to 128479.10024659803. For more clarity, we will use variables for the parameters of the function.

$$y = -128390.42104270413 \times e^{-0.0000001587477443985961 7\times x} + 128479.1002465980 = a \times e^{b\times x} + c$$

If we set a limit for $x \to \infty$ then:

$$y = \lim_{x\to\infty} a \times e^{bx} + c$$

Remember that $e^{-\infty} = 0$.

$$y = \cancel{a \times e^{-b\infty}} + c \implies y = c$$

That means that equation 3 falls asymptotically towards 128479.1002465980. Note that this is the best-fit model of numerical computation of the first $1,000,000$ positive integers and not a model for the infinitude of prime numbers.

B Appendix: Comparing algorithms

For the enhancement of the computational performance, we reduced the iterations per integer in the range to $\sqrt{x}$. In figure 5 we show the diagram comparing the logical way (iterating each positive integer smaller than x) and the short time-memory saving algorithm. Both models are identical in finding the distribution of multiplication for the first 100 positive integers.

C Appendix: Derivation of angle equation

Having two linear functions (we have converted the exponential equation into a linear slope), we can compute the angles of the slopes relative to the x-axis.

$$y_1 = m_1 \times x + c$$

$$y_2 = m_2 \times x + d$$

$$\theta_1 = \arctan m_1$$

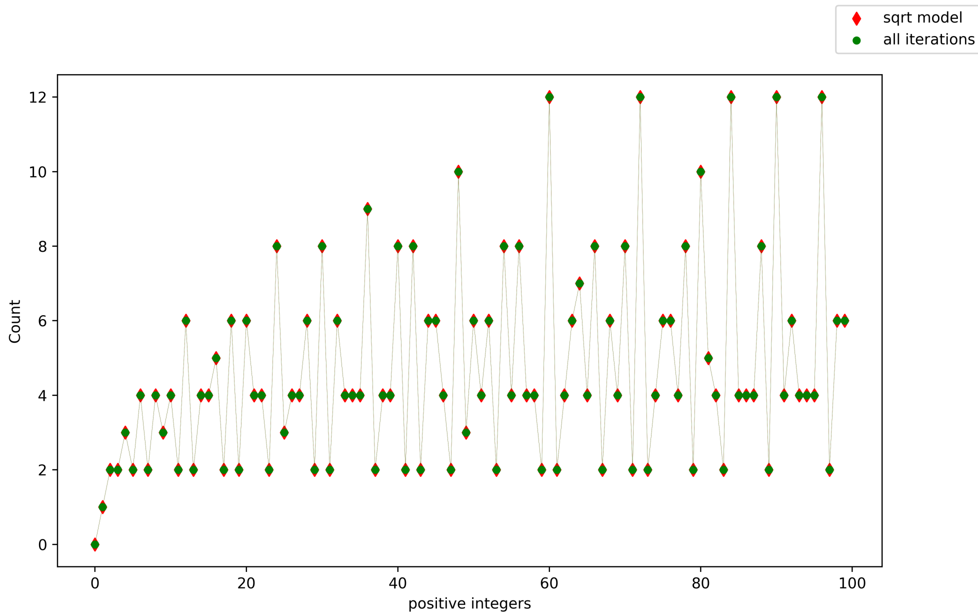

Figure 5: Here the all-iterations model is shown in green, while the $\sqrt{x}$ model is plotted in red. Both models are identical in finding the distribution of multiplication for the first 100 positive integers.

$$\theta_2 = \arctan m_2$$

The angle between the two slopes is the difference of the angle between the slopes and the x-axis:

$$\theta = \mid \arctan \theta_1 - \arctan \theta_2 \mid$$

Then we can use the trigonometric identity to solve this for θ:

$$\tan(\theta_1 - \theta_2) = \tan\theta = \frac{\tan\theta_1 - \tan\theta_2}{1 + \tan\theta_1 \tan\theta_2}$$

As we know:

$$\tan\theta_1 = m_1 \quad \text{and} \quad \tan\theta 2 = m_2$$

$$\implies \tan\theta = \frac{\mid m_1 - m_2 \mid}{1 + m_1 m_2} \tag{6}$$

Then we come to our formula:

$$\theta = \arctan \frac{\mid m_1 - m_2 \mid}{1 + m_1 m_2}$$

References

[AD13] F Aydin and G Dogan. Development of a new integer hash function with variable length using prime number set. *Balkan Journal of Electrical & Computer Engineering*, 1(1):10–14, 2013.

[Bri24] Encyclopedia Britannica. "prime", 5 2024.

[Den04] Lih-Yuan Deng. Generalized mersenne prime number and its application to random number generation. In *Monte Carlo and Quasi-Monte Carlo Methods 2002: Proceedings of a Conference held at the National University of Singapore, Republic of Singapore, November 25–28, 2002*, pages 167–180. Springer, 2004.

[DLVP96] Charles De La Vallee-Poussin. Recherches analytiques sur la théorie des nombres premiers. *Ann. Soc. Sc. Bruxelles*, 1896.

[Had96] Jacques Hadamard. Sur la distribution des zéros de la fonction $zeta(s)$ et ses conséquences arithmétiques. *Bulletin de la Societé mathematique de France*, 24:199–220, 1896.

[HD07] John Hunter and Darren Dale. The matplotlib user's guide. *Matplotlib 0.90. 0 user's guide*, 2007.

[HMVDW+20] Charles R Harris, K Jarrod Millman, Stéfan J Van Der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J Smith, et al. Array programming with numpy. *Nature*, 585(7825):357–362, 2020.

[HW79] Godfrey Harold Hardy and Edward Maitland Wright. *An introduction to the theory of numbers.* Oxford university press, 1979.

[MSP+17] Aaron Meurer, Christopher P. Smith, Mateusz Paprocki, Ondřej Čertík, Sergey B. Kirpichev, Matthew Rocklin, AMiT Kumar, Sergiu Ivanov, Jason K. Moore, Sartaj Singh, Thilina Rathnayake, Sean Vig, Brian E. Granger, Richard P. Muller, Francesco Bonazzi, Harsh Gupta, Shivam Vats, Fredrik Johansson, Fabian Pedregosa, Matthew J. Curry, Andy R. Terrel, Štěpán Roučka, Ashutosh Saboo, Isuru Fernando, Sumith Kulal, Robert Cimrman, and Anthony Scopatz. Sympy: symbolic computing in python. *PeerJ Computer Science*, 3:e103, January 2017.

[RSA78] Ronald L Rivest, Adi Shamir, and Leonard Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[VGO+20] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.