TFY4235 - The World of Quantum Mechanics

Karl Kristian Ladegård Lockert[a]

[a] *Institutt for fysikk, Norges Teknisk-Naturvitenskapelige Universitet, N-7491 Trondheim, Norway.*

Abstract

Abstract

1. Introduction

$$i\hbar\frac{\partial\psi}{\partial t} = H\psi \tag{1}$$

$$H\psi_n = E_n\psi_n \tag{2}$$

Throughout the report, quantities with a bar ($\bar{A}$) that may have an analytic counterpart (A), will represent quantities which are (to be) computed.

2. Quantum mechanics in a box

The Schrödinger equation tells us the time-evolution of the wave function, which, according to the Copenhagen interpretation of quantum mechanics, has the physical interpretation of a probability amplitude when squaring the absolute value. In our current setup, where we have

$$V = \begin{cases} 0, & 0 < x < L \\ \infty, & \text{otherwise} \end{cases}, \tag{3}$$

there is 0% chance of finding the particle inside the "walls" (at $x < 0$ or $x > L$). Thus, the wave function must go to zero at these points. Defining $x' = x/L$ and $t' = t/t_0$, and inserting in Equation (1).

$$\begin{aligned} i\hbar\frac{\partial\psi}{\partial t} &= i\hbar\frac{\partial\psi}{\partial t'}\frac{\partial t'}{\partial t} = \frac{-\hbar^2}{2mL^2}\frac{\partial^2\psi}{\partial x'^2} \\ \implies i\frac{\partial\psi}{\partial t'} &= \frac{-\hbar}{2mL^2\frac{\partial t'}{\partial t}}\frac{\partial\psi}{\partial x'^2} \end{aligned}$$

Thus, setting

$$t' = \frac{\hbar}{2mL^2}t, \quad x' = \frac{x}{L} \tag{4}$$

gives the wanted dimensionless equation

$$i\frac{\partial\psi}{\partial t'} = -\frac{\partial^2\psi}{\partial x'^2}. \tag{5}$$

Inserting our new variables in Equation (2) with

$$H = \frac{\hat{p}^2}{2m} + V(x) = -\frac{-\hbar}{2mL^2}\frac{\partial^2}{\partial x'^2} + \tilde{V}(x'),$$

we get

$$\begin{aligned} \frac{-\hbar^2}{2mL^2}\frac{\partial^2\psi_n}{\partial x'^2} &= E_n\psi_n \implies \\ -\frac{\partial^2\psi_n}{\partial x'^2} &= \lambda_n\psi_n, \end{aligned} \tag{6}$$

with the relation

$$\lambda_n = \frac{2mL^2E_n}{\hbar^2} \tag{7}$$

between the energy levels and the dimensionless eigenvalues. The boundary conditions that the wave function disappears in the walls, but the walls are now at $x' = 0$ and $x' = 1$. It is now clear that choosing $x_0 = L$ is suitable, as it makes us able to work on the simple domain $[0, 1]$. Any other proportionality constant $\alpha \in \mathbb{R}$ such that $x' = \alpha x/L$ should work as well, scaling both the time and energies by a factor α^2, as long as x' is dimensionless. Other scaling possibilities are also possible, and have consequences for the analytic expressions for ψ. For example scaling the interval to be mirrored about $x = 0$ would only pick out the even terms in a Fourier series expansion.

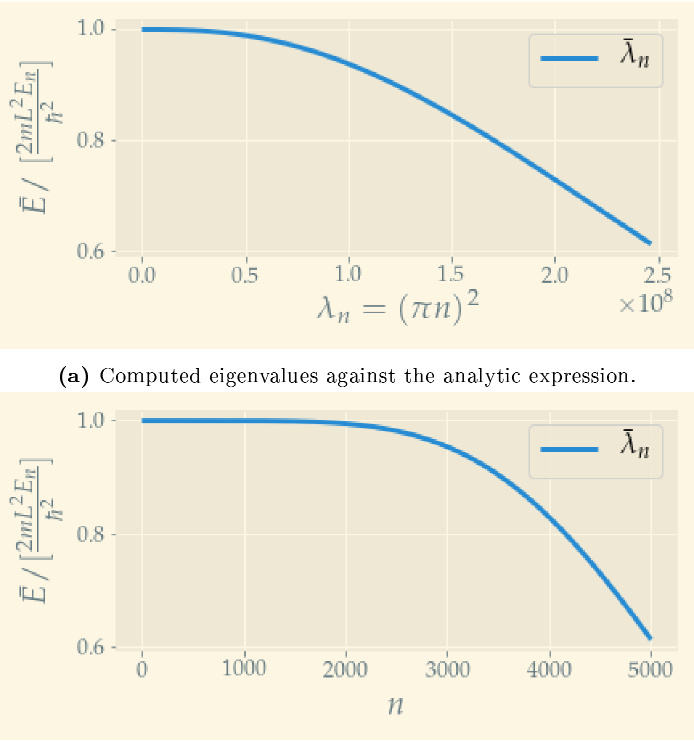
Solving Equation (6) with the imposed boundary conditions can be done analytically in the following manner. Since we are looking for solutions that have self-similar second derivatives with an extra minussign, we guess a solution on the form $\psi = A_n\sin\big(\sqrt{\lambda_n}x'\big) + B_n\cos\big(\sqrt{\lambda_n}x'\big)$. The boundary condition $\psi(x' = 0) = 0$ gives $B_n = 0$, while the boundary condition $\psi(x' = 1) = 0$ gives the restriction to λ_n that $\sqrt{\lambda_n} = n\pi$, hence the labels n are also justified. Our analytic solution is therefore

$$\psi_n(x') = \mathcal{N}\sin(\pi n x'), \tag{8}$$

where $\mathcal{N}$ is a normalization constant to be decided.

$$\begin{aligned} 1 = \langle\psi_n|\psi_n\rangle &= \mathcal{N}^2\int_0^1 \mathrm{d}x' \sin^2(n\pi x') \\ &= \mathcal{N}^2\int_0^1 \mathrm{d}x' \,\frac{1-\cos(2\pi n x')}{2} = \frac{\mathcal{N}^2}{2} \\ &\implies \mathcal{N} = \sqrt{2}, \end{aligned}$$

and we have the analytic solution as announced in [1].



(a) Computed eigenvalues against the analytic expression.

(b) Computed eigenvalues against n.

Figure 1: Comparison of computed eigenvalues and the analytic expressions.

The implementation of a finite-difference-scheme is done using sparse matrix formatting for the cases when the discretization step is very small. Otherwise a dense matrix format is sufficient for solving eigenvalues, using the "eigsh"-function from NumPy's linear algebra library.

In Figure 1 a comparison of the calculated eigenvalues against both the analytic expression for λ_n and n is shown for the case where the interval is discretized in 5000 points. Notice that $\bar{\lambda}_n/\lambda_n$ is close to 1 for n up to about 2000, where the energies are 2000^2 times higher than the ground state.

To compute the error of a numerical solution, we must have a metric of some sort. Let us introduce

$$E[\bar{\psi}_n(x')] = \int \mathrm{d}x' \, ||\bar{\psi}_n(x')| - |\psi_n(x')||^2 \tag{9}$$

as an example, where the bar denotes the numerical approximation to the analytic function. Then, a comparison of the eigenvectors with respect to the number of discretization steps can be made. In Section 2, this is shown for the three lowest energy levels.

The implementation of

$$\alpha_n = \langle \psi_n | \Psi_0 \rangle = \int \mathrm{d}x' \, \psi_n^*(x') \Psi_0(x') \tag{10}$$

can be done in a simple fashion by taking optimized inner-product implementations for general vectors, so calculating the actual integral is actually not needed. In Figure 3, the inner product of the first 100 eigenfunctions are shown, and indicates orthogonality of the states. The off-diagonals are ~ 15 orders of magnitude smaller than the diagonal, which we might attribute to numerical artifacts or the imperfectness of the solutions, which we have seen is present, c.f. Figure 1 and Section 2.

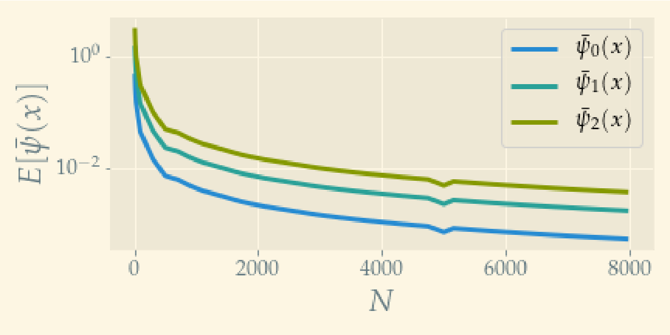


Figure 2: Error of the ground state, first- and second excited states as a function of the discretization steps.

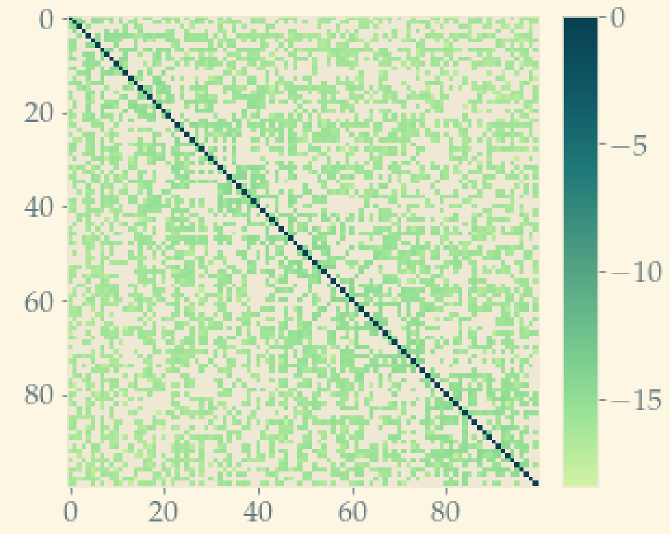


Figure 3: Logarithm (base 10) of the inner product of the 100 lowest excited eigenfunctions.

Using the reduced units previously discussed, we can express the full state

$$\Psi(x,t) = \sum_n \alpha_n \exp\left(-\frac{iE_n t}{\hbar}\right) \psi_n(x) \tag{11}$$

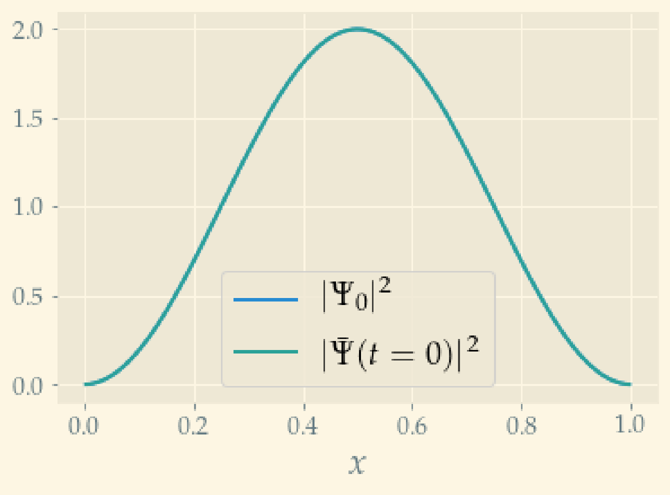


Figure 4: Analytic and computed probability density of initial state Ψ_0 using $N = 1000$ as the number of discretization points

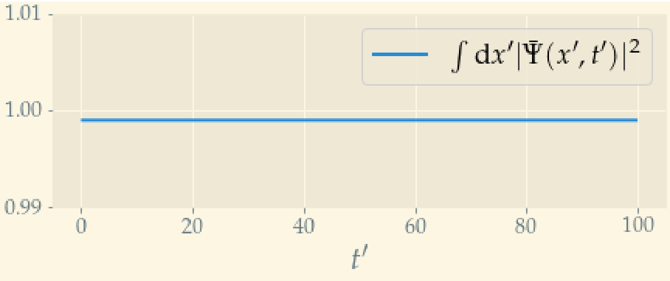


Figure 5: Normalization of the computed wave function for the time interval $t' \in [0, 100]$ width $N = 10000$ and $\Psi_0(x') = \delta(x' - \frac{1}{2})$.

as

$$\sum_n \alpha_n \exp\left(-\frac{iE_n t}{\hbar}\right)\psi_n(x) = \sum_n \alpha_n \exp\left(-\frac{iE_n t}{\hbar}\right)\psi_n(x)$$

$$= \sum_n \exp\left(-\frac{i\frac{\hbar^2\lambda_n}{2mL^2}\frac{2mL^2 t'}{\hbar}}{\hbar}\right)\psi_n(x')$$

$$\implies \Psi(x', t') = \sum_n \alpha_n \exp(-i\lambda_n t')\psi_n(x'). \quad (12)$$

Now using the ground state

$$\Psi_0 = \sqrt{2}\sin(\pi x) \quad (13)$$

as initial condition, we can compute the full state. In Figure 4 the initial states are plotted both for the analytic expression and the computed value at $t = 0$. The normalization is also well-defined, as seen in Figure 5. For Ψ_0 as in Equation (13), this is expected, since Ψ_0 is orthogonal to $\Psi_{i\neq 0}$, and Ψ_0 is properly normalized itself. For $\Psi_0 = \delta(x' - 1/2)$, however, this is more complicated. In this case, $c_n = \int \mathrm{d}x'\, \psi_n^* \delta(x' - 1/2)$ gives $c_n = \psi_n$, which means that only half of the eigenfunctions in Equation (8) will contribute. Numerically, however, we can only estimate the value of the Dirac-delta with a finite number of plane waves, in contrast to its integral representation $\delta \sim \int \mathrm{d}k\, \mathrm{e}^{ikx}$.

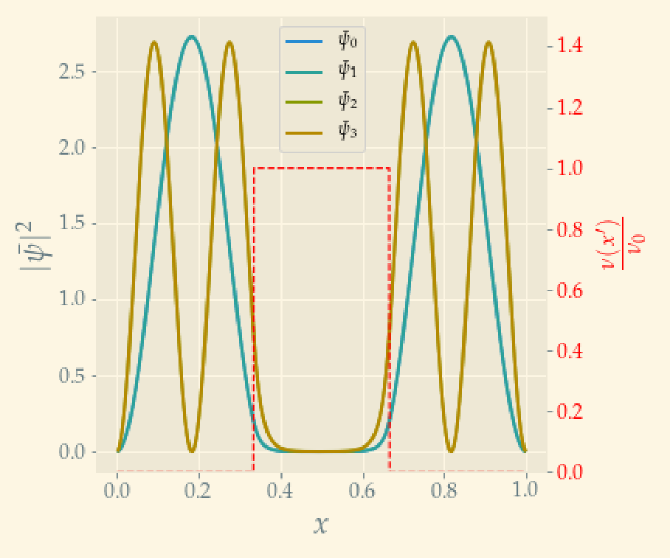


Figure 6: Degeneracy of the ground state and first excited states plotted together with the potential.

3. Adding a barrier to the box-potential

Let us now consider a potential barrier in the box, modeled by the dimensionless potential $\nu(x') = t_0 V_0/\hbar$, where $t_0 = \frac{2mL^2}{\hbar}$ as in Equation (4), and is given by

$$\nu(x') = \begin{cases} 0, & 0 < x' < \frac{1}{3} \\ \nu_0 = \frac{2mL^2 V_0}{\hbar^2}, & \frac{1}{3} < x' < \frac{2}{3} \\ 0, & \frac{2}{3} < x' < 1 \\ \infty, & \text{otherwise} \end{cases} \quad (14)$$

Thus, setting ν_0 to 0 is the same task as we had for the previous case, since we have the same boundary conditions and a vanishing potential between the edges, which ensures that the matrix-representation (and thus the eigenvalues) of the operator acting on the wave function is equal. Now we set $\nu_0 = 10^3$. Comparing the energy eigenvalues for the problem, we find that some seem to be equal, i.e. there seems to be degeneracy in the system. This cannot be, since we are in 1D with a discrete energy spectrum [2]. Indeed, the states are equal, as we can see for the lowest lying energy eigenfunctions in Figure 6. We now prepare the initial state

$$\Psi_0 = \frac{1}{\sqrt{2}}\left(\psi_1(x') + \psi_2(x')\right) \quad (15)$$

and let it evolve from $t'_0 = 0$ to

$$t'_1 = \frac{\pi}{\lambda_2 - \lambda_1}, \quad (16)$$

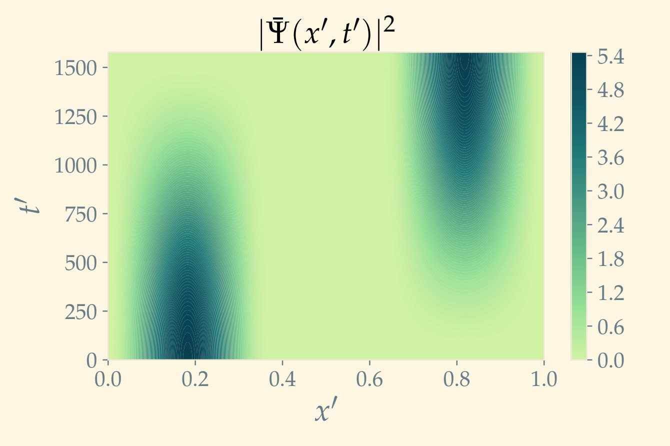


Figure 7: Tunneling across the potential barrier.

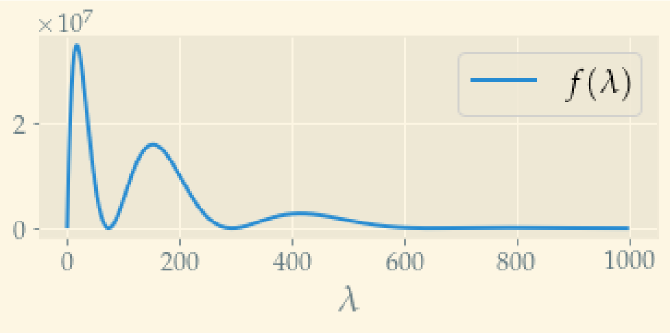


Figure 8: The analytic expression for finding the eigenvalues.

we find that the particle has tunneled from one side of the barrier to the other! This is shown in Figure 7. We can see that the initial state (on the left side of the potential barrier) "leaks" into the opposite side as time progress. Inserting Equation (16) into Equation (12), we find (after some manipulation)

$$\Psi(x', t_1') = C\left(\psi_2(x') - \psi_1(x')\right) \tag{17}$$

for a normalization constant C. This is the initial state mirrored about $x' = 1/2$.

3.1. Root finding

Following ref. [1], the eigenvalues of the analytic solutions are given by the roots of the function $f(\lambda)$, shown in Figure 8. This is in good agreement with the computed eigenvalues. By using SciPy's [3] optimized root-finding algorithm with the computed eigenvalues as the initial guess, we find three distinct eigenvalues below $\nu_0 = 10^3$. Using the implementation of the root finder, the roots of $f(\lambda)$ is found more precicely, and the first couple of roots are shown in Table 1. Investigating the roots of $f(\lambda)$ for different values of ν_0, we find that the value of ν_0 that separates having one and no states with $\lambda < \nu_0$ is for

$$\nu_0 = 22.10526(3). \tag{18}$$

Computed λ	Roots of $f(\lambda)$
73.49662578	73.93560016
73.49861656	
291.77230762	293.49231502
291.79660664	
644.62162712	648.26437316
645.10279446	

Table 1: A comparison of computed eigenvalues and computed roots of the analytic expression for $f(\lambda)$. The units are given by Equation (7).

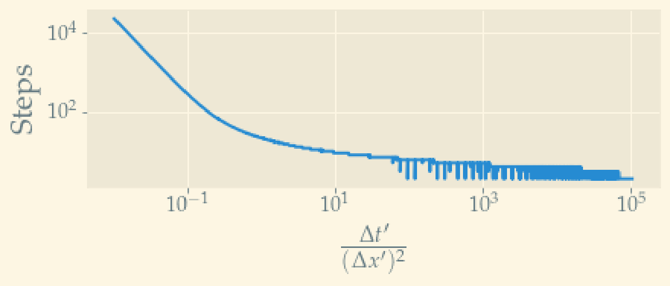


Figure 9: The number of steps taken before the Wave Function breaks down against the Courant-Friedrichs-Lewy (CFL)-number on a log-log scale.

4. Step-by-step time evolution

For a time-dependent Hamiltonian, an expansion in stationary eigenstates does not work in the same simple fashion. This is because the states ψ_i are eigenvectors of the *instantaneous* Hamiltonian. For different times , these vectors are not any more eigenvectors of the full Hamiltonian. That being said, with a step-by-step implementation we should get rid of the apparent degeneracy of the eigenvalues.

4.1. Euler scheme

We now implement the Euler scheme for evaluating its applicability for the quantum problem. An interesting phenomena occur, where the function abruptly breaks down after a number of time steps. The approach to finding the number of steps before this breakdown is by looking at the normalization of the wave function. As this suddenly diverges, we can use this as a check in a while-loop for comparing step sizes in both temporal and spatial direction.

In Figure 9 the steps before the breakdown of the simulation for the Euler-Scheme is shown. This suggests that for the a successful simulation in this scheme, we need $\Delta t' \ll (\Delta x')^2$. At thee same time we also want $\Delta x' \ll 1$, so this would require a very heavy computation. A better approach is to use a different numerical scheme.

4.2. Crank-Nicolson

The implementation of this scheme is done by computing the LU-decomposition of $\left(1 + \frac{i}{2}\Delta t' \hat{H}\right)$ and solving a

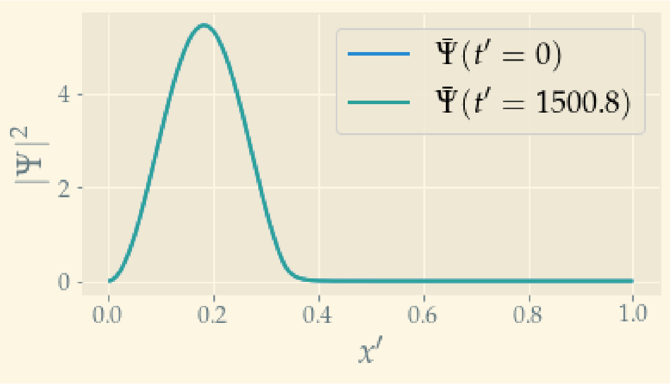


Figure 10: Initial state and time-evolved state using the Crank-Nicolson scheme.

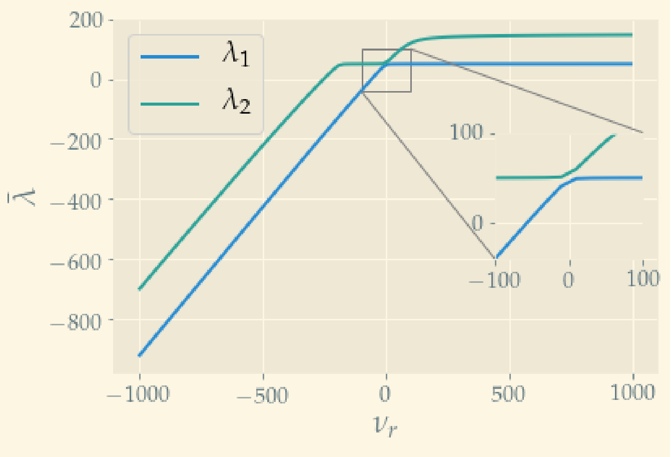


Figure 11: The two lowest eigenvalues plotted against the varying potential ν_r.

linear system $A\mathbf{x} = b$ for each time step. As previously mentioned, we should get rid of the degeneracy of states. With this in mind, the phase $\sim \frac{1}{\lambda_2 - \lambda_1}$ should diverge, and we have to propagate the system infinitely in time. By preparing the same state as in Equation (15), we can use the Crank-Nicolson scheme to propagate the function and check this. As viewed in Figure 10, there is now no tunneling, as opposed to the case when we project the states on a plane-wave basis, c.f. Figure 7.

4.3. Two level system

Figure 11 shows the two lowest lying energy eigenvalues for $V(x,t)$ as introduced in ref.[1]. For $\nu_r = 0$, the energy difference is $\varepsilon_0 \simeq 5.6962$.

The expectation value

$$\tau = \langle g_0|\hat{H}|e_0\rangle \tag{19}$$

can be computed as the previous inner products, applying first the action of $\hat{H}$ to $|e_0\rangle$ and then taking the inner product. τ is found to take a linear shape

$$\tau(\nu_r) \simeq -0.429\nu_r \tag{20}$$

References

[1] J. T. Kjellstadli, A. Sala, and I. Simonsen. Assignment 3: The world of quantum mechanics. `http://web.phys.ntnu.no/~ingves/Teaching/TFY4235/Assignments/TFY4235_Assignment_03.pdf`, 2020.

[2] Per Christian Hemmer. *Kvantemekanikk*. Tapir Akademiske Forlag, 5 edition, 2005.

[3] 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, CJ Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake Vand erPlas, 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.