

Physics 411: Homework 5

This homework is due the first Tuesday after the break, **Tuesday March 11**.

1. **The QR algorithm:** In this exercise you'll write a program to calculate the eigenvalues and eigenvectors of a real symmetric matrix using the QR algorithm. The first challenge is to write a program that finds the QR decomposition of a matrix. Then we'll use that decomposition to find the eigenvalues.

As we discussed in class, the QR decomposition expresses a real square matrix \mathbf{A} in the form $\mathbf{A} = \mathbf{QR}$, where \mathbf{Q} is an orthogonal matrix and \mathbf{R} is an upper-triangular matrix. Given an $N \times N$ matrix \mathbf{A} we can compute the QR decomposition as follows.

Let us think of the matrix as a set of N column vectors $\mathbf{a}_0 \dots \mathbf{a}_{N-1}$ thus:

$$\mathbf{A} = \begin{pmatrix} | & | & | & \cdots \\ \mathbf{a}_0 & \mathbf{a}_1 & \mathbf{a}_2 & \cdots \\ | & | & | & \cdots \end{pmatrix},$$

where we have numbered the vectors in Python fashion, starting from zero, which will be convenient when writing the program. We now define two new sets of vectors $\mathbf{u}_0 \dots \mathbf{u}_{N-1}$ and $\mathbf{q}_0 \dots \mathbf{q}_{N-1}$ as follows:

$$\begin{aligned} \mathbf{u}_0 &= \mathbf{a}_0, & \mathbf{q}_0 &= \frac{\mathbf{u}_0}{|\mathbf{u}_0|}, \\ \mathbf{u}_1 &= \mathbf{a}_1 - (\mathbf{q}_0 \cdot \mathbf{a}_1)\mathbf{q}_0, & \mathbf{q}_1 &= \frac{\mathbf{u}_1}{|\mathbf{u}_1|}, \\ \mathbf{u}_2 &= \mathbf{a}_2 - (\mathbf{q}_0 \cdot \mathbf{a}_2)\mathbf{q}_0 - (\mathbf{q}_1 \cdot \mathbf{a}_2)\mathbf{q}_1, & \mathbf{q}_2 &= \frac{\mathbf{u}_2}{|\mathbf{u}_2|}, \end{aligned}$$

and so forth. The general formulas for calculating \mathbf{u}_i and \mathbf{q}_i are

$$\mathbf{u}_i = \mathbf{a}_i - \sum_{j=0}^{i-1} (\mathbf{q}_j \cdot \mathbf{a}_i)\mathbf{q}_j, \quad \mathbf{q}_i = \frac{\mathbf{u}_i}{|\mathbf{u}_i|}.$$

- (a) Show, by induction or otherwise, that the vectors \mathbf{q}_i are orthonormal, i.e., that they satisfy

$$\mathbf{q}_i \cdot \mathbf{q}_j = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j. \end{cases}$$

Now, rearranging the definitions of the vectors, we have

$$\begin{aligned} \mathbf{a}_0 &= |\mathbf{u}_0| \mathbf{q}_0, \\ \mathbf{a}_1 &= |\mathbf{u}_1| \mathbf{q}_1 + (\mathbf{q}_0 \cdot \mathbf{a}_1)\mathbf{q}_0, \\ \mathbf{a}_2 &= |\mathbf{u}_2| \mathbf{q}_2 + (\mathbf{q}_0 \cdot \mathbf{a}_2)\mathbf{q}_0 + (\mathbf{q}_1 \cdot \mathbf{a}_2)\mathbf{q}_1, \end{aligned}$$

and so on. Or we can group the vectors \mathbf{q}_i together as the columns of a matrix and write all of these equations as a single matrix equation

$$\mathbf{A} = \begin{pmatrix} | & | & | & \cdots \\ \mathbf{a}_0 & \mathbf{a}_1 & \mathbf{a}_2 & \cdots \\ | & | & | & \cdots \end{pmatrix} = \begin{pmatrix} | & | & | & \cdots \\ \mathbf{q}_0 & \mathbf{q}_1 & \mathbf{q}_2 & \cdots \\ | & | & | & \cdots \end{pmatrix} \begin{pmatrix} |\mathbf{u}_0| & \mathbf{q}_0 \cdot \mathbf{a}_1 & \mathbf{q}_0 \cdot \mathbf{a}_2 & \cdots \\ 0 & |\mathbf{u}_1| & \mathbf{q}_1 \cdot \mathbf{a}_2 & \cdots \\ 0 & 0 & |\mathbf{u}_2| & \cdots \end{pmatrix}.$$

(If this looks complicated it's worth multiplying out the matrices on the right to verify for yourself that you get the correct expressions for the \mathbf{a}_i .)

Notice now that the first matrix on the right-hand side of this equation, the matrix with columns \mathbf{q}_i , is orthogonal, because the vectors \mathbf{q}_i are orthonormal, and the second matrix is upper triangular. In other words, we have found the QR decomposition $\mathbf{A} = \mathbf{QR}$. The matrices \mathbf{Q} and \mathbf{R} are

$$\mathbf{Q} = \begin{pmatrix} | & | & | & \cdots \\ \mathbf{q}_0 & \mathbf{q}_1 & \mathbf{q}_2 & \cdots \\ | & | & | & \cdots \end{pmatrix}, \quad \mathbf{R} = \begin{pmatrix} |\mathbf{u}_0| & \mathbf{q}_0 \cdot \mathbf{a}_1 & \mathbf{q}_0 \cdot \mathbf{a}_2 & \cdots \\ 0 & |\mathbf{u}_1| & \mathbf{q}_1 \cdot \mathbf{a}_2 & \cdots \\ 0 & 0 & |\mathbf{u}_2| & \cdots \end{pmatrix}.$$

- (b) Write a Python function that takes as its argument a real square matrix \mathbf{A} and returns the two matrices \mathbf{Q} and \mathbf{R} that form its QR decomposition. As a test case, try out your function on the matrix

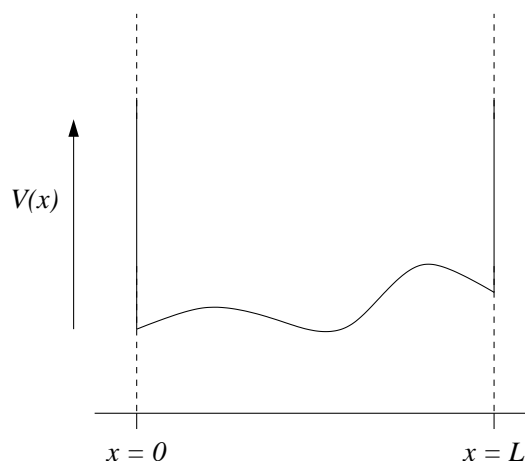
$$\mathbf{A} = \begin{pmatrix} 1 & 4 & 8 & 4 \\ 4 & 2 & 3 & 7 \\ 8 & 3 & 6 & 9 \\ 4 & 7 & 9 & 2 \end{pmatrix}.$$

Check the results by multiplying \mathbf{Q} and \mathbf{R} together to recover the original matrix \mathbf{A} again.

- (c) Using your function, write a complete program to calculate the eigenvalues and eigenvectors of a real symmetric matrix using the QR algorithm. Continue the calculation until the magnitude of every off-diagonal element of the matrix is smaller than 10^{-6} . Test your program on the example matrix above. You should find that the eigenvalues are 1, 21, -3 , and -8 .

✓ **For full credit** turn in your proof from part (a), a printout of your program, and a printout if it in action, showing the results it gives for the test matrix.

2. **Asymmetric quantum well:** Quantum mechanics can be formulated as a matrix problem and solved on a computer using linear algebra methods. Suppose, for example, we have a particle of mass M in a one-dimensional quantum well of width L :



But it's not a square well like the examples you've probably seen before. Suppose instead that the potential $V(x)$ varies somehow inside the well, as sketched in the figure. We cannot solve such problems analytically in general, but we can solve them on the computer.

In a pure state of energy E , the spatial part of the wavefunction obeys the time-independent Schrödinger equation $\hat{H}\psi(x) = E\psi(x)$, where the Hamiltonian operator \hat{H} is given by

$$\hat{H} = -\frac{\hbar^2}{2M} \frac{d^2}{dx^2} + V(x).$$

For simplicity, let's assume that the walls of the well are infinitely high, so that the wavefunction is zero outside the well, which means it must go to zero at $x = 0$ and $x = L$. In that case, the wavefunction can be expressed as a Fourier sine series thus:

$$\psi(x) = \sum_{n=1}^{\infty} \psi_n \sin \frac{\pi nx}{L},$$

where ψ_1, ψ_2, \dots are the Fourier coefficients.

(a) Noting that, for m, n positive integers

$$\int_0^L \sin \frac{\pi mx}{L} \sin \frac{\pi nx}{L} dx = \begin{cases} L/2 & \text{if } m = n, \\ 0 & \text{otherwise,} \end{cases}$$

show that the Schrödinger equation $\hat{H}\psi = E\psi$ implies that

$$\sum_{n=1}^{\infty} \psi_n \int_0^L \sin \frac{\pi mx}{L} \hat{H} \sin \frac{\pi nx}{L} dx = \frac{1}{2} LE \psi_m.$$

Hence, defining a matrix \mathbf{H} with elements

$$\begin{aligned} H_{mn} &= \frac{2}{L} \int_0^L \sin \frac{\pi mx}{L} \hat{H} \sin \frac{\pi nx}{L} dx \\ &= \frac{2}{L} \int_0^L \sin \frac{\pi mx}{L} \left[-\frac{\hbar^2}{2M} \frac{d^2}{dx^2} + V(x) \right] \sin \frac{\pi nx}{L} dx, \end{aligned}$$

show that Schrödinger's equation can be written in matrix form as $\mathbf{H}\boldsymbol{\psi} = E\boldsymbol{\psi}$, where $\boldsymbol{\psi}$ is the vector (ψ_1, ψ_2, \dots) . Thus $\boldsymbol{\psi}$ is an eigenvector of the *Hamiltonian matrix* \mathbf{H} with eigenvalue E . If we can calculate the eigenvalues of this matrix, then we know the allowed energies of the particle in the well.

(b) For the case $V(x) = ax/L$, evaluate the integral in H_{mn} analytically and so find a general expression for the matrix element H_{mn} . Show that the matrix is real and symmetric. You'll probably find it useful to know that

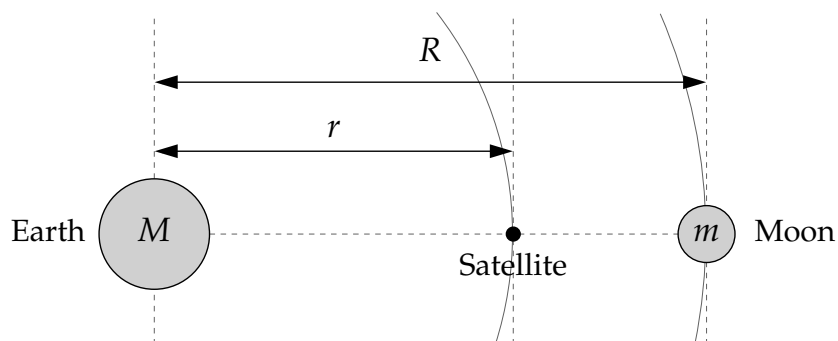
$$\int_0^L x \sin \frac{\pi mx}{L} \sin \frac{\pi nx}{L} dx = \begin{cases} 0 & \text{if } m \neq n \text{ and both even or both odd,} \\ -\left(\frac{2L}{\pi}\right)^2 \frac{mn}{(m^2 - n^2)^2} & \text{if } m \neq n \text{ and one is even, one is odd,} \\ L^2/4 & \text{if } m = n. \end{cases}$$

Write a Python function to evaluate your expression for H_{mn} for arbitrary m and n when the particle in the well is an electron, the well has width 5 \AA , and $a = 10 \text{ eV}$. (The mass and charge of an electron are $9.1094 \times 10^{-31} \text{ kg}$ and $1.6022 \times 10^{-19} \text{ C}$ respectively.)

- (c) The matrix \mathbf{H} is in theory infinitely large, so we cannot represent it exactly on the computer. But we can get a pretty accurate solution for the first few eigenvalues by cutting off the matrix after the first few elements. Using the function you wrote for part (b) above, write a program to create a 10×10 array of the elements of \mathbf{H} up to $m, n = 10$. Calculate the eigenvalues of this matrix using the appropriate function from `numpy.linalg` and hence print out, in units of electron volts, the first ten energy levels of the quantum well, within this approximation. You should find, for example, that the ground-state energy of the system is around 5.84 eV . (Hint: Bear in mind that matrix indices in Python start at zero, while the indices in standard algebraic expressions, like those above, start at one. You will need to make allowances for this in your program.)
- (d) Modify your program to use a 100×100 array instead and again calculate the first ten energy eigenvalues. Comparing with the values you calculated in part (c), what do you conclude about the accuracy of the calculation?

✓ **For full credit** turn in your derivations from parts (a) and (b), a printout of your final program from part (c), your results from parts (c) and (d), and your conclusions from part (d).

3. **The Lagrange point:** There is a magical point between the Earth and the Moon, called the L_1 Lagrange point, at which a satellite will orbit the Earth in perfect synchrony with the Moon, staying always in between the two. This works because the inward pull of the Earth and the outward pull of the Moon combine to create exactly the needed centripetal force that keeps the satellite in its orbit. Here's the setup:



- (a) Assuming circular orbits, and assuming that the Earth is much more massive than either the Moon or the satellite, show that the distance r from the center of the Earth to the L_1 point satisfies

$$\frac{GM}{r^2} - \frac{Gm}{(R-r)^2} = \omega^2 r,$$

where M and m are the Earth and Moon masses, G is Newton's gravitational constant, and ω is the angular velocity of both the Moon and the satellite.

- (b) The equation above is a fifth-order polynomial equation in r (also called a quintic equation). Such equations cannot be solved exactly in closed form, but it's straightforward to solve them numerically. Write a program to solve for the distance r from the Earth to the L_1 point. You can use any of the methods we have studied—the relaxation method, binary search, Newton's method, and the secant method all work well for this problem. Compute a solution accurate to at least four significant figures. The values of the various parameters are:

$$G = 6.674 \times 10^{-11} \text{ m}^3 \text{ kg}^{-1} \text{ s}^{-2},$$

$$M = 5.974 \times 10^{24} \text{ kg},$$

$$m = 7.348 \times 10^{22} \text{ kg},$$

$$R = 3.844 \times 10^8 \text{ m},$$

$$\omega = 2.662 \times 10^{-6} \text{ s}^{-1}.$$

You will also need to choose a suitable starting value for r , or two starting values if you use the binary search or secant methods.

✓ **For full credit** turn in a copy of your program and your result for the distance r .

4. **The temperature of a light bulb:** An incandescent light bulb is a simple device—it contains a resistive filament, usually made of tungsten, heated by the flow of electricity until it becomes hot enough to radiate thermally. Essentially all of the power consumed by such a bulb is radiated as electromagnetic energy, but some of the radiation is not in the visible wavelengths, which means it is useless for lighting purposes.

Let us define the efficiency of a light bulb to be the fraction of the radiated energy that falls in the visible band. It's a good approximation to assume that the radiation obeys the Planck radiation law, meaning that the power radiated per unit wavelength λ obeys

$$I(\lambda) = 2\pi Ahc^2 \frac{\lambda^{-5}}{e^{hc/\lambda k_B T} - 1},$$

where A is the surface area of the filament, T is the temperature, h is Planck's constant, c is the speed of light, and k_B is Boltzmann's constant. The visible wavelengths run from $\lambda_1 = 390 \text{ nm}$ to $\lambda_2 = 750 \text{ nm}$, so the total energy radiated in the visible window is $\int_{\lambda_1}^{\lambda_2} I(\lambda) d\lambda$ and the total energy at all wavelengths is $\int_0^\infty I(\lambda) d\lambda$. Dividing one expression by the other and substituting for $I(\lambda)$ from above, we get an expression for the efficiency η of the light bulb thus:

$$\eta = \frac{\int_{\lambda_1}^{\lambda_2} \lambda^{-5} / (e^{hc/\lambda k_B T} - 1) d\lambda}{\int_0^\infty \lambda^{-5} / (e^{hc/\lambda k_B T} - 1) d\lambda},$$

where the leading constants and the area A have canceled out. Making the substitution $x = hc/\lambda k_B T$, this can also be written as

$$\eta = \frac{\int_{hc/\lambda_2 k_B T}^{hc/\lambda_1 k_B T} x^3 / (e^x - 1) dx}{\int_0^{\infty} x^3 / (e^x - 1) dx} = \frac{15}{\pi^4} \int_{hc/\lambda_2 k_B T}^{hc/\lambda_1 k_B T} \frac{x^3}{e^x - 1} dx,$$

where we have made use of the known exact value of the integral in the denominator.

- (a) Write a Python function that takes a temperature T as its argument and calculates the value of η for that temperature from the formula above. The integral in the formula cannot be done analytically, but you can do it numerically using any method of your choice. (For instance, Gaussian quadrature with 100 sample points works fine.) Use your function to make a graph of η as a function of temperature between 300 K and 10 000 K. You should see that there is an intermediate temperature where the efficiency is a maximum.
- (b) Calculate the temperature of maximum efficiency of the light bulb to within 1 K using golden ratio search. What efficiency does the bulb achieve at this temperature?
- (c) Is it practical to run a tungsten-filament light bulb at the temperature you found? If not, why not?

☑ **For full credit** turn in a printout of your final (golden-ratio search) program, your plot from part (a), and your results and discussion from parts (b) and (c).