Computational Quantum Physics
Exercise 1

PHYS-463
Prof. G. Carleo

Problem 1.1 The perils of calculating derivatives

Suppose we have a function $f(x)$ and we want to calculate its derivative at a point x. We can do that with pencil and paper if we know the mathematical form of the function, or we can do it on the computer by making use of the definition of the derivative:

$$\frac{\mathrm{d}f}{\mathrm{d}x} = \lim_{\delta \to 0} \frac{f(x+\delta) - f(x)}{\delta}. \tag{1}$$

On the computer we can't actually take the limit as δ goes to zero, but we can get a reasonable approximation just by making δ small.

1. Write a program that defines a function $f(x)$ returning the value $x(x-1)$, then calculates the derivative of the function at the point $x = 1$ using the formula above with $\delta = 10^{-2}$. Calculate the true value of the same derivative analytically and compare with the answer your program gives. The two will not agree perfectly. Why not?

2. Repeat the calculation for $\delta = 10^{-4}, 10^{-6}, 10^{-8}, 10^{-10}, 10^{-12}, 10^{-14}$, and 10^{-16}. You should see that the accuracy of the calculation initially gets better as δ gets smaller, but then gets worse again. Why is this?

Problem 1.2 Game of Life

Conway's Game of Life is a zero-player game, which means that besides an initial configuration, the game does not require any additional inputs. The game works as follows: On a 2D grid there are cells that are alive (1) and cells that are dead (0). The cells change their status according to the following set of rules:

- If a living cell has less than 2 or more than 3 living neighbours, it dies (under-/over population).
- If a living cell has 2 or 3 living neighbours, it stays alive.
- If a dead cell has exactly 3 living neighbours, it becomes alive (reproduction).

In the following you will write a program that implements the Conway's Game of Life:

1. Write a method that takes as an input a configuration of cells and gives as an output another configuration of cells, updated according to the above rules (for cells at the border, the neighbours that fall out of the grid are all considered dead).

2. Use your method to study and evolve a system of size (30×30) with initial configuration
$[(15,15),(15,16),(15,17),(16,16)]$ (here (x,y) are the x and y coordinates of the initial living cells) over 100 update steps. Save the state of each update step and use the template to make an animation of the evolution. You should see a stable pattern emerging. Next try $[(10,1),(10,5),(11,6),(11,5),(10,2),(10,3),(11,3),(10,4)]$ over 200 update steps, what can you observe? Take a look at http://pi.math.cornell.edu/~lipa/mec/lesson6.html for further special initializations.

* Game of Life is Turing Complete meaning that with a suitable initial pattern, one can do any computation (Turing Machine)

Problem 1.3 Diagonalizing random hermitian matrices

In this problem we study the spectrum of a particular family of random hermitian matrices called Gaussian unitary ensemble (GUE) through exact diagonalization.

The Gaussian unitary ensemble is a probability distribution over the space of complex $n \times n$ hermitian matrices $\{H \in \mathbb{C}^{n\times n} : H_{ij} = H^\star_{ji}\}$ with probability density

$$p(H) = \frac{1}{Z} e^{-\frac{n}{2}\operatorname{tr}\left[H^2\right]} \tag{2}$$

where the normalization Z is chosen such that the integral over the whole space is 1.

It is possible to generate samples from this distribution by generating a complex matrix A with entries sampled independently from a standard complex normal distribution

$$A_{ij} \sim \mathcal{CN}(0, 1) \tag{3}$$

and taking

$$H_{ij} = \frac{1}{\sqrt{2n}}(A_{ij} + A^\star_{ji}) \tag{4}$$

for the hermitian matrix.
Here we denote with $\mathcal{CN}(0, 1)$ the standard complex normal distribution with mean 0 and variance 1 which is given by $p(z) \propto e^{-|z|^2}$ for $z \in \mathbb{C}$. A derivation for this can be found in appendix A.
Note that we can generate samples $Z \sim \mathcal{CN}(0, 1)$ by taking the real and imaginary part independently from a (real) normal distribution with mean 0 and variance $\frac{1}{2}$, i.e. $\mathrm{Re}[Z], \mathrm{Im}[Z] \sim \mathcal{N}(0, \frac{1}{2})$ where $\mathcal{N}(0, \sigma^2)$ is the usual real normal distribution with probability density $p(x) \propto e^{-\frac{1}{2\sigma^2}x^2}$.

$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$ Variance σ^2

1. Write a function which takes the size n as input and returns a random $n \times n$ GUE Matrix

2. Generate a random hermitian GUE matrix of size $n = 2048$ with it and numerically compute it's full spectrum (the set of all eigenvalues $\{\lambda_i\}$). spectrum = np.linalg.eigvalsh (H)

3. Plot the density of the distribution of eigenvalues (using a histogram).

   We expect it to follow the Wigner semicircle law

   $$p(\lambda) = \begin{cases} \frac{2}{\pi R^2}\sqrt{R^2 - \lambda^2} & \text{if } |\lambda| \le R \\ 0 & \text{if } |\lambda| > R \end{cases} \tag{5}$$

   with the radius $R = 2$ in our case.

4. Generate 100000 random hermitian matrices of size $n = 4$ from the GUE.

   Compute the mean $\langle H_{ij}\rangle$ and the correlation function $\langle H_{ij}H^\star_{mn}\rangle$. Verify that the mean is 0 and that the correlation function is equal to

   $$\langle H_{ij}H^\star_{mn}\rangle = \frac{1}{n}\delta_{im}\delta_{jn} \tag{6}$$

A Sampling from the Gaussian unitary Ensemble

For completeness we briefly sketch the derivation for the procedure in Problem 1.3 for sampling from the Gaussian unitary ensemble

$$p(H) = \frac{1}{Z} e^{-\frac{n}{2}\operatorname{tr}\left[H^2\right]} \tag{7}$$

over the hermitian matrices $\{H \in \mathbb{C}^{n\times n} : H_{ij} = H_{ji}^{\star}\}$
We start by noting that trace of the square of a hermitian matrix can be expressed in terms of the norm of the entries of the diagonal and upper triangular part:

$$\operatorname{tr}\left[H^2\right] = \sum_i H_{ii}^{\star} H_{ii} + 2\sum_{i<j} H_{ij}^{\star} H_{ij} = \sum_i |H_{ii}|^2 + 2\sum_{i<j} |H_{ij}|^2 \tag{8}$$

This means we can write eq. (7) as a product of independent gaussians in the following way:

$$p(H) \propto e^{-\frac{n}{2}\operatorname{tr}\left[H^2\right]} = \prod_i e^{-\frac{n}{2}|H_{ii}|^2} \prod_{i<j} e^{-n|H_{ij}|^2} \tag{9}$$

Since the diagonal entries H_{ii} of the hermitian matrix are real we therefore have that $H_{ii} \sim \mathcal{N}(0, 1/n)$ for the diagonal entries and similarly $H_{ij} \sim \mathcal{CN}(0, 1/n)$ for the entries in the upper triangle. The entries in the lower triangle are then given by the hermitian transpose of the upper triangle. Using basic facts about the addition of two normal random variables and multiplication of one with a constant we see that taking $A_{ij} \sim \mathcal{CN}(0, 1)$, summing $A_{ij} + A_{ji}^{\star}$ and scaling it with $\frac{1}{\sqrt{2n}}$ gives a hermititan matrix whose entries follow the desired distributions.

Problem 1.1 The perils of calculating derivatives

Suppose we have a function $f(x)$ and we want to calculate its derivative at a point x. We can do that with pencil and paper if we know the mathematical form of the function, or we can do it on the computer by making use of the definition of the derivative:

$$\frac{\mathrm{d}f}{\mathrm{d}x} = \lim_{\delta\to 0} \frac{f(x+\delta) - f(x)}{\delta}. \tag{1}$$

On the computer we can't actually take the limit as δ goes to zero, but we can get a reasonable approximation just by making δ small.

1. Write a program that defines a function $f(x)$ returning the value $x(x-1)$, then calculates the derivative of the function at the point $x = 1$ using the formula above with $\delta = 10^{-2}$. Calculate the true value of the same derivative analytically and compare with the answer your program gives. The two will not agree perfectly. Why not?

2. Repeat the calculation for $\delta = 10^{-4}, 10^{-6}, 10^{-8}, 10^{-10}, 10^{-12}, 10^{-14}$, and 10^{-16}. You should see that the accuracy of the calculation initially gets better as δ gets smaller, but then gets worse again. Why is this?

1. Since we used finite δ.

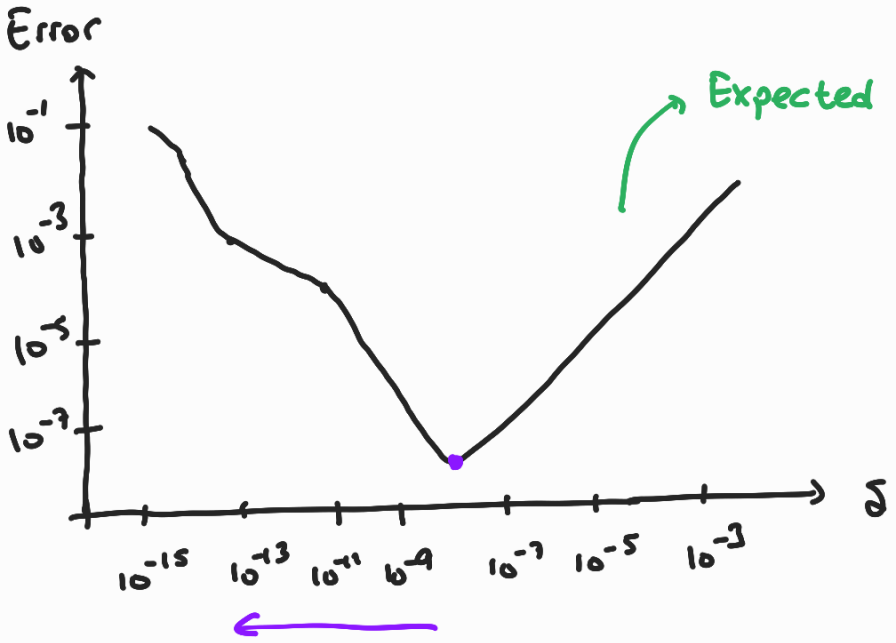
Error can be quantitatively using Taylor expansion:

$$\frac{f(x+\delta) - f(x)}{\delta} = \frac{f(x) + f'(x)\delta + \frac{1}{2}f''(x)\delta^2 + O(\delta^3) - f(x)}{\delta}$$

$$= f'(x) + \frac{1}{2}f''(x)\delta + O(\delta^2)$$

For $f(x) = x(x-1) \rightarrow f'''(x) = 0$, error only contains linear term δ.

2)



why this occurs?

"floating-point error"

$$e_{tot}(\delta) = \frac{f(x+\delta+\epsilon)-f(x)}{\delta} - f'(x)$$

$$= \frac{f(x+\epsilon) + f'(x+\epsilon)\delta + \frac{1}{2}f''(x+\epsilon)\delta^2 + O(\delta^3) - f(x)}{\delta} - f'(x)$$

$$= \frac{f(x+\epsilon)-f(x)}{\delta} + f'(x+\epsilon) - f'(x) + \frac{1}{2}f''(x+\epsilon)\delta + O(\delta^2)$$

$$\left(f'(x) = \frac{f(x+\epsilon)-f(x)}{\epsilon} = f(x+\epsilon)-f(x) = \epsilon f'(x) \right)$$

$$= \frac{f'(x)\epsilon}{\delta} + \frac{1}{2}f''(x)\delta + O(\delta^2) + O(\epsilon)$$

$e_{tot}(\delta)$ is minimized when δ is of the order $\sqrt{\epsilon}$.

For this case, float is floatG with 53-bit significant precision, $\epsilon \approx 2^{-53} \approx 10^{-16} \rightarrow e_{tot}$ minimized when $\delta \approx 10^{-8}$

Problem 1.3 Diagonalizing random hermitian matrices

In this problem we study the spectrum of a particular family of random hermitian matrices called Gaussian unitary ensemble (GUE) through exact diagonalization.

The Gaussian unitary ensemble is a probability distribution over the space of complex $n \times n$ hermitian matrices $\{H \in \mathbb{C}^{n\times n} : H_{ij} = H^\star_{ji}\}$ with probability density

$$p(H) = \frac{1}{Z} e^{-\frac{n}{2}\operatorname{tr}[H^2]} \tag{2}$$

where the normalization Z is chosen such that the integral over the whole space is 1.

It is possible to generate samples from this distribution by generating a complex matrix A with entries sampled independently from a standard complex normal distribution

$$A_{ij} \sim \mathcal{CN}(0, 1) \tag{3}$$

and taking

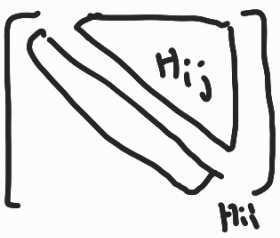
$$H_{ij} = \frac{1}{\sqrt{2n}}(A_{ij} + A^\star_{ji}) \tag{4}$$

for the hermitian matrix.
Here we denote with $\mathcal{CN}(0, 1)$ the standard complex normal distribution with mean 0 and variance 1 which is given by $p(z) \propto e^{-|z|^2}$ for $z \in \mathbb{C}$. A derivation for this can be found in appendix A.
Note that we can generate samples $Z \sim \mathcal{CN}(0, 1)$ by taking the real and imaginary part independently from a (real) normal distribution with mean 0 and variance $\frac{1}{2}$, i.e. $\operatorname{Re}[Z], \operatorname{Im}[Z] \sim \mathcal{N}(0, \frac{1}{2})$ where $\mathcal{N}(0, \sigma^2)$ is the usual real normal distribution with probability density $p(x) \propto e^{-\frac{1}{2\sigma^2}x^2}$.

$p(H) = \frac{1}{Z} e^{-n/2\ \mathrm{tr}[H^2]}$ where $H \in C^{n\times n} : H_{ij} = H_{ji}^*$

$\mathrm{tr}[H^2] = \sum_i H_{ii}^* H_{ii} + 2\sum_{i<j} H_{ij}^* H_{ij} = \sum_i |H_{ii}|^2 + 2\sum_{i<j} |H_{ij}|^2$

$p(H) \propto e^{-\frac{n}{2}\mathrm{tr}[H^2]} = \prod_i e^{-\frac{n}{2}|H_{ii}|^2} + \prod_{i<j} e^{-n|H_{ij}|^2}$



$H_{ii} \sim N(0, 1/n)$
real

$H_{ij} \sim CN(0, 1/n)$