

Fifth Assignment for Computational Physics

Xinyu Liu

October 8, 2024

Contents

1	My Github Page URL	2
2	Problem 1: Exercise 5.15 of Newman	2
3	Problem 2: Exercise 5.17 of Newman, Gamma function	3
4	Problem 3: SVD for data fitting	5

1 My Github Page URL

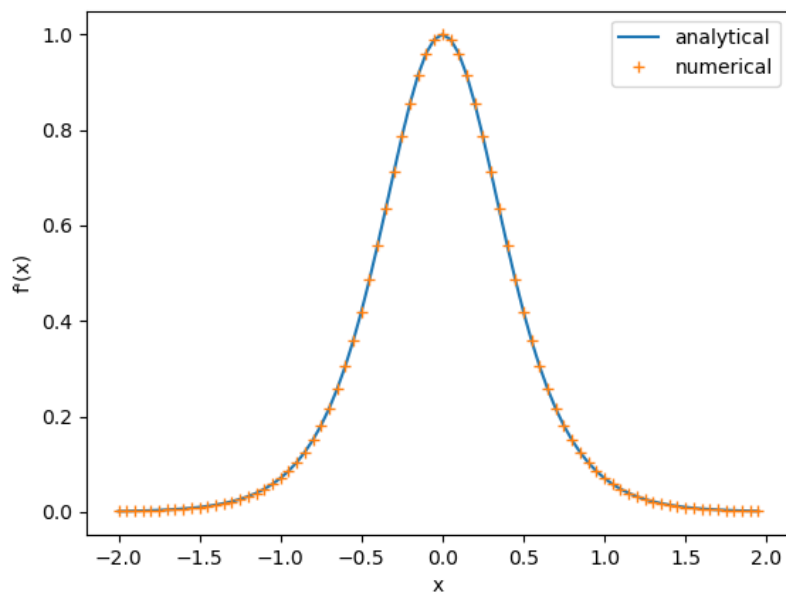
<https://github.com/rising1227/phys-ga2000>

2 Problem 1: Exercise 5.15 of Newman

The Associated Code for this part is ps5-1.py.

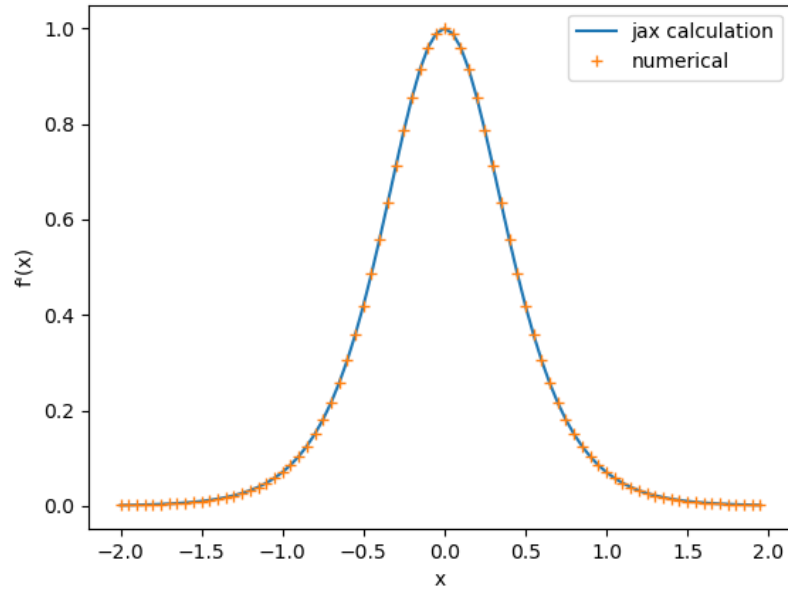
We made a function $f(x) = 1 + 0.5 * \tanh(2x)$ in the program and find both the analytical derivative $f'(x) = \frac{1}{np.cosh(2x)^2}$ and numerical derivative of that function. The result is plotted as follows:

Table 1: Numerical and analytical derivative of a function



We also tried the python package jax to calculate the derivative. It work exactly as advertised:

Table 2: Numerical and jax derivative of a function

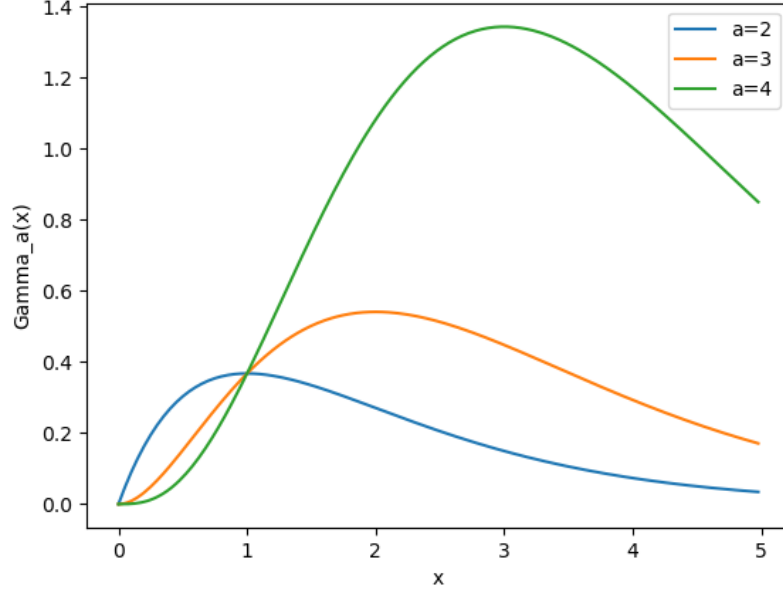


3 Problem 2: Exercise 5.17 of Newman, Gamma function

The Associated Code for this part is ps5-2.py.

We do exactly expected from the problem. First we plotted the integrand inside Gamma function with $a = 2, 3, 4$.

Table 3: Integrand for Gamma function with different a



With $f(x) = x^{a-1}e^{-x}$, the derivative of this function is:

$$f'(x) = (a-1)x^{a-2}e^{-x} - x^{a-1}e^{-x} = (a-1-x)x^{a-2}e^{-x} \quad (1)$$

So when $x = a-1$, the derivative is 0 and the function takes its maximum value.

Then we do the mapping from infinite integration range to (0,1). For different a, we choose the mapping function to be:

$$g(x) = \frac{x}{(a-1) + x} \quad (2)$$

Also to avoid doing multiplication between very large and small number, we use the formula:

$$x^{a-1} = e^{(a-1)\ln(x)} \quad (3)$$

So that the integrand for Gamma function becomes:

$$e^{(a-1)\ln(x)-x} \quad (4)$$

This is helpful because either x^{a-1} will be extremely large and e^{-x} will be extremely small for large x , thus inducing large computation error.

After proper change of expression and mapping the integration range, the final integration we'll do is:

$$\int_0^1 \frac{a-1}{(1-x)^2} * e^{-(a-1)*x/(1-x)+(a-1)*\ln((a-1)*x/(1-x))} dx \quad (5)$$

We use gaussian quadrature to numerically evaluate this integral. The integration result is the same as expected:

Calculation of Gamma[1.5] is 0.8862269613087213

Calculation of Gamma[3] is 2.0000000000000004

Calculation of Gamma[6] is 119.99999999999999

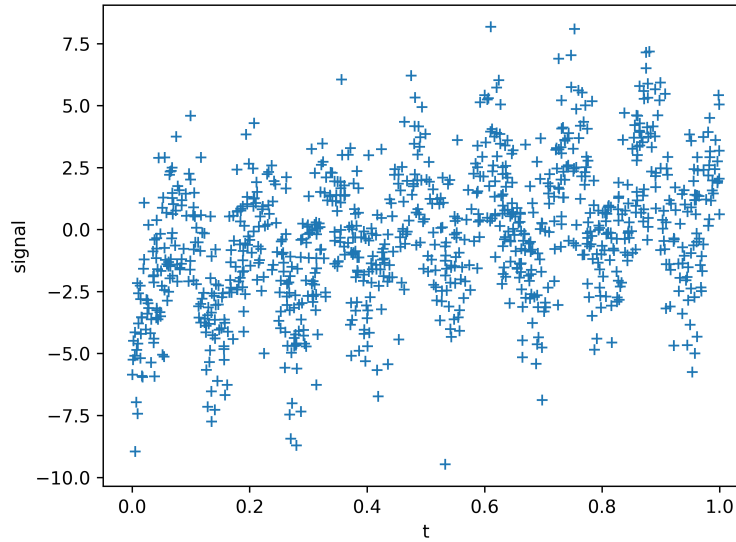
Calculation of Gamma[10] is 362879.9999999999

4 Problem 3: SVD for data fitting

The Associated Code for this part is ps5-3.py.

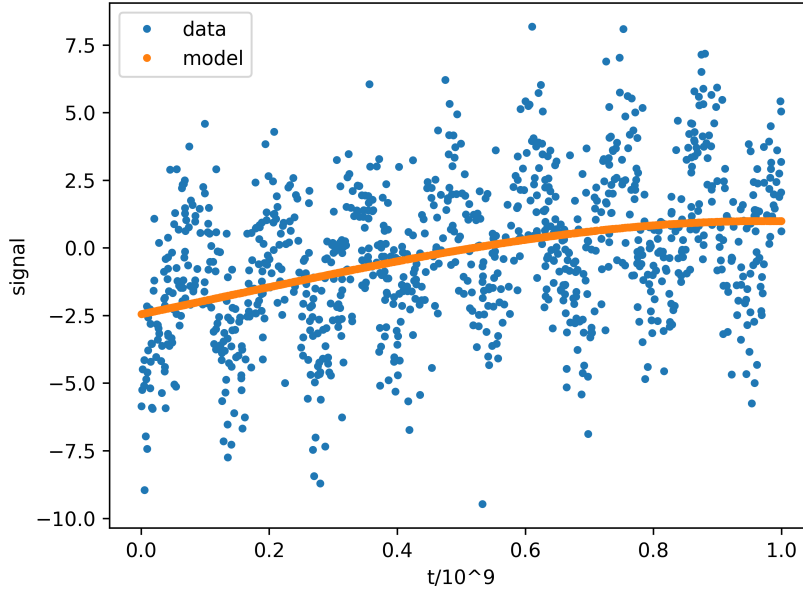
We extract the signal data and time and first plot it out:

Table 4: Signal data against time



In order to avoid calculating very large number, we first rescale the time data by $t' = t * 10^{-9}$. Then, we use SVD to fit the data to third-order polynomial. Here's our result:

Table 5: Fitted signal data to third-order polynomial



Then we calculated the residual of our observation data. We especially calculated the number of observation data being inside one sigma of uncertainty. We found that there's only 0.573 portion of the data inside one sigma, which is significantly lower than the expected portion of 0.68 by gaussian distribution. We think that this cannot satisfies us.

We then attempt to fit the data with 10th-order polynomial and 20th-order polynomial. Here's our fitted result:

Table 6: Fitted signal data to 10th-order polynomial

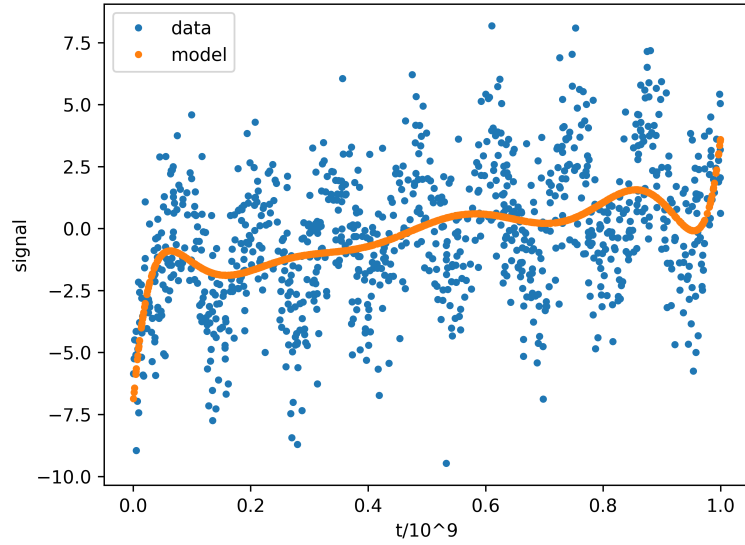
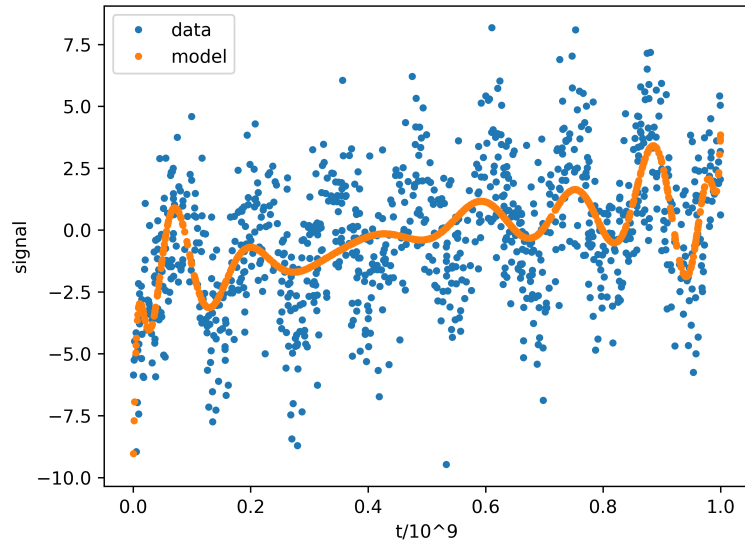


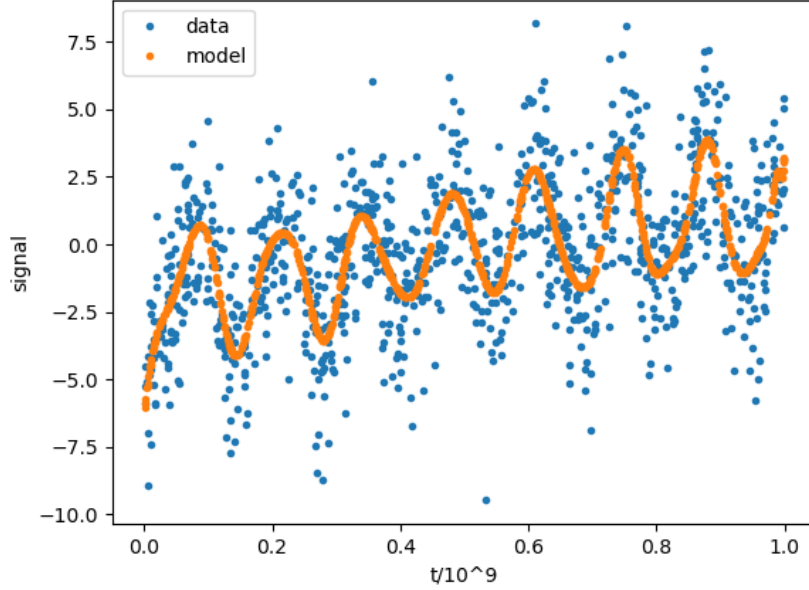
Table 7: Fitted signal data to 20th-order polynomial



The condition number is respectively 3898066.283726428(10th order) and 154815469309063.1(20th order). We found that the fitted result is not good enough and the condition number has become insanely big.

Finally we tried to fit the data with function with routine(doin fourier decomposition). We fit our data with sine and cosine function with frequency from 3.0 to 8.0, with step of 0.25. Here's our fitted result:

Table 8: Fitted signal data to routine functions



We found that this fitting result lead to 0.694 of all datas inside one sigma for theoretical prediction, which is close to theoretical result of 0.683 for gaussian distribution. This implies that our model is doing a quite good job explaining the observation data. We can actually look at the fitted diagram or go back to find the maximum value of singular value inside W to see that the principal frequency is around $7.5(*10^9)$