Universidad del País Vasco

Facultad de Ciencia y Tecnología

NSDE: Practice 4

Group 2

Nerea Andia

Ander Cano

Mikel Lamela

Aitor Larrinoa

March 2021

Computer practice 4

4.1 The continuous and the discrete Fourier transform

Statement

(a) Write a program in *Mathematica* that computes the first, second and third derivatives, using the commands `Fourier` and `InverseFourier` with the option `FourierParameters -> {1, -1}`.

- Run the program, for $N = 8$, $N = 16$ and $N = 64$, for the functions

$$f(x) = \exp(\sin(x)), \qquad f(x) = \exp(\sin(x)^3) \qquad x \in [0, 2\,\pi] \tag{1}$$

- Plot the original function and its derivatives computed using the *Mathematica* command `Derivative` or `D`, or better with the ' (prime) command.
- In a plot, compare the exactly computed derivatives with the approximated ones, obtained using the DFT.
- Compute the error (absolute error) of the computed derivatives.

Solution

We are asked to write a program in *Mathematica* that computes the first, second and third derivatives of two different functions and we have to solve it using different values of N (Number of points of the grid). These values are 8, 16 and 64.

This 4.1 part will consist of two sections. The first section will be related with the $\mathrm{f}_1(\mathrm{x}) = \exp(\sin(x))$ function, and the second part with $\mathrm{f}_2(\mathrm{x}) = \exp(\sin(x)^3)$. Each section will contain the solution with the different values of N mentioned before. This way it will be easier to compare the results obtained taking into account the number of points in each section.

0.0.1 1. function $f_1(x) = \exp(\sin(x))$

On the one hand, we have plotted the original function and its derivatives. Since the statement tells us that it is better to use the '(prime) command rather than Derivative or D commands, we have used the '(prime) one. This plot is independent of the value of N we are considering because we are just plotting f_1 and its first, second and third derivatives. The result obtained is the next one:

On the other hand, we are asked to compare the exactly computed derivatives with the approximated ones obtained using DFT.

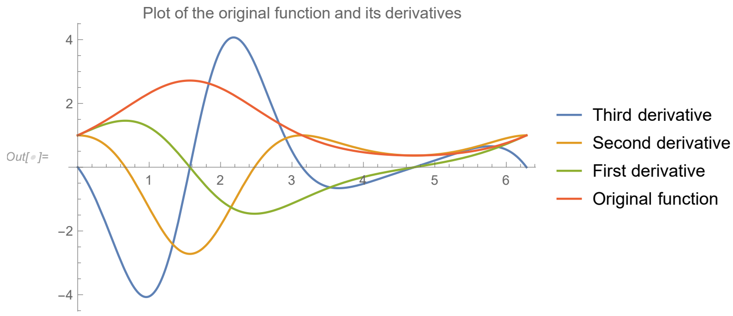


Figure 1: Plot of the original $f_1(x)$ function and its derivatives

FIRST DERIVATIVE

Next, we will plot the approximate solution of the first derivative and the real solution of the first derivative. The approximated solution has been calculated by the Discrete Fourier Transform (DFT). We will compare the approximation just mentioned with the different values of N mentioned before. We will start with $N = 8$.

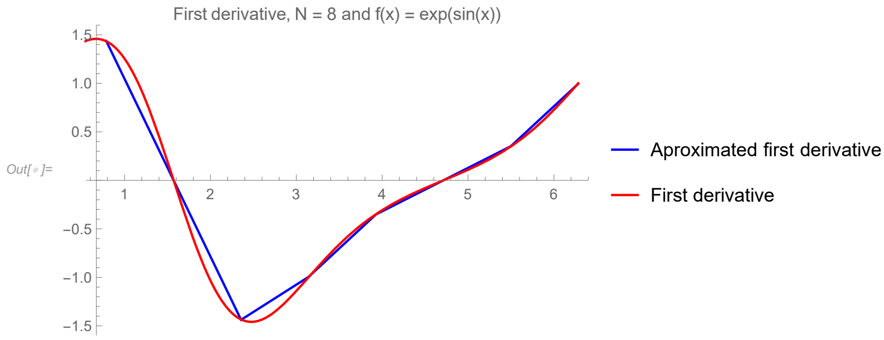


Figure 2: Plot of the comparison of the first derivative of $f_1(x)$ using DFT and *Mathematica* command ' (prime) for $N = 8$

We continue now with $N = 16$.

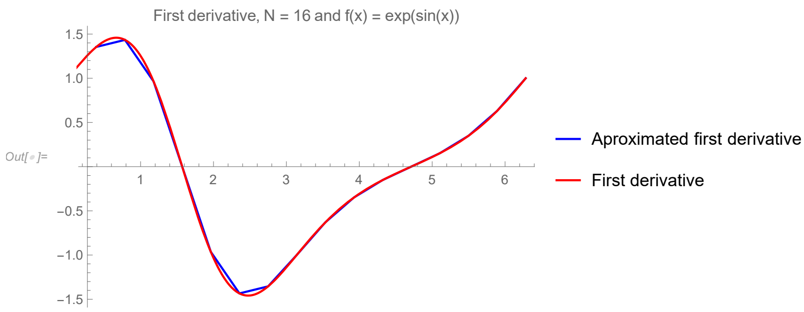


Figure 3: Plot of the comparison of the first derivative of $f_1(x)$ using DFT and *Mathematica* command ' (prime) for $N = 16$.

And we finish with N = 64.

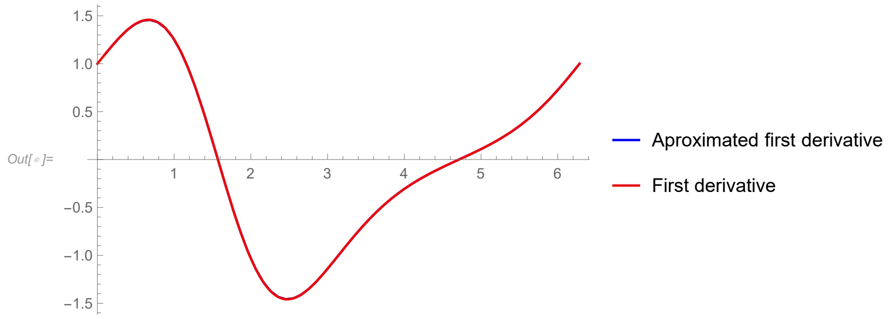


Figure 4: Plot of the comparison of the first derivative of $f_1(x)$ using DFT and *Mathematica* command ' (prime) for N = 64.

We can see that as N is bigger, the approximation becomes better. For N = 8, the approximation is poor at the humps, for N = 16 the approximation begins to look a little bit smoother and for N = 64 we can say the approximation is good. Thus, when taking N (number of points) $\geq 2^4$, we can say that the approximation will be good enough. It is important to mention that the number of points has to be a power of 2.

SECOND DERIVATIVE

Now, we will plot the approximate solution of the second derivative and the real solution of it comparing the different values of N. The approximated solution has been calculated making use of DFT.

We are starting first with N = 8.

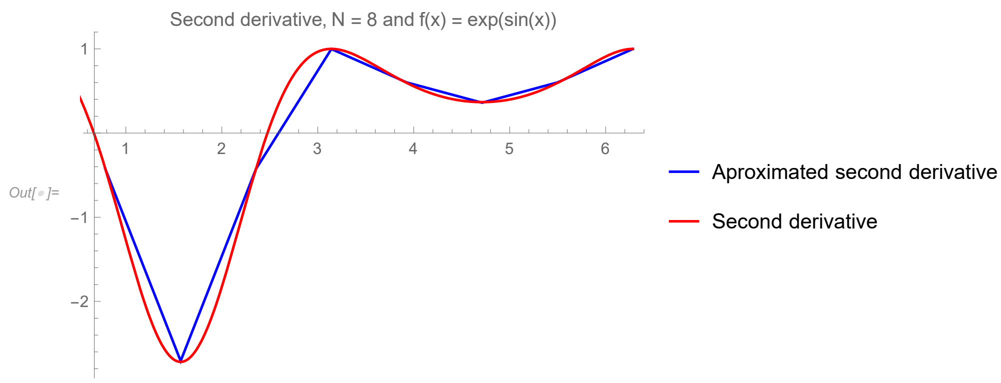


Figure 5: Plot of the comparison of the second derivative of $f_1(x)$ using DFT and *Mathematica* command ' (prime) for N = 8.

We continue plotting the one related with N = 16

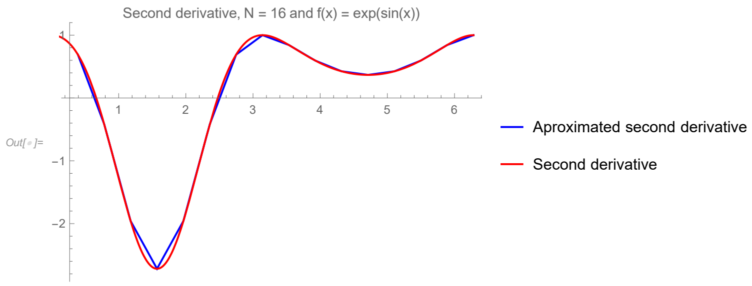


Figure 6: Plot of the comparison of the second derivative of $f_1(x)$ using DFT and *Mathematica* command ' (prime) for N = 16.

And we finish plotting the one with N = 64.

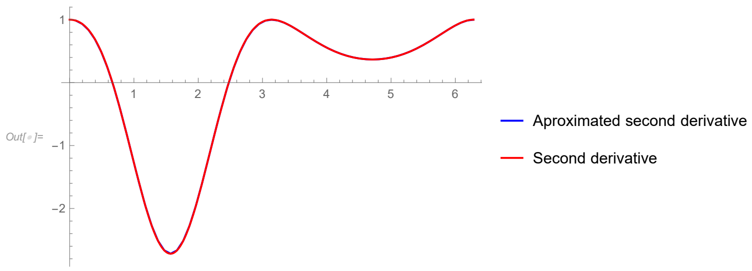


Figure 7: Plot of the comparison of the second derivative of $f_1(x)$ using DFT and *Mathematica* command ' (prime) for N = 64.

As we have seen for the first derivative, the approximation of the second derivative becomes better when the value of N is bigger. Indeed, considering 64 number of points in the grid, the approximation is quite good. Then, we can conclude that when $N \geq 2^4$, we get a good approximation for the second derivative.

THIRD DERIVATIVE

As we have done for the first and second derivative, we need to compare the results obtained with the approximated solution of the third derivative obtained using DFT. We will consider the different values of N as we have done for the first and second derivative.

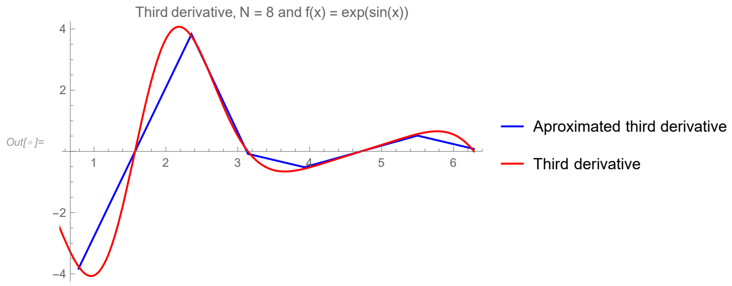


Figure 8: Plot of the comparison of the third derivative of $f_1(x)$ using DFT and *Mathematica* command ' (prime) for N = 8.

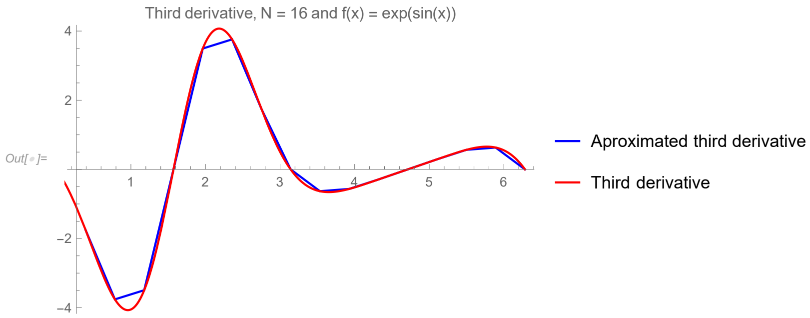


Figure 9: Plot of the comparison of the third derivative of $f_1(x)$ using DFT and Mathematica command ’ (prime) for N = 16.

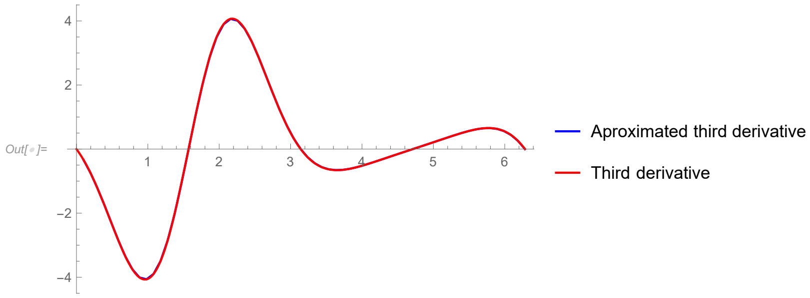


Figure 10: Plot of the comparison of the third derivative of $f_1(x)$ using DFT and Mathematica command ’ (prime) for N = 64.

In this case, we can also see that the approximation is smoother when N = 16, than it is when N = 8, and that it is smoother when N = 64 than it is when N =16. In fact, the N = 8 approximation is very poor in all aspects, while the approximation with N = 16 is poor at the corners. Thus, we can conclude that the approximation becomes better when N is bigger, just like first and second derivatives.

Apart from comparing the approximated and real solution of the derivatives, we are also asked to compute the absolute error of the computed derivatives. On the one hand, we will put the values for different N in a table:

Values	N = 8	N = 16	N = 64
First derivative	0.00431791	1.76189×10^{-7}	8.84015×10^{-15}
Second derivative	0.0102933	3.90952×10^{-7}	2.55213×10^{-13}
Third derivative	0.0814325	1.17951×10^{-5}	4.35823×10^{-12}

As we can see, bigger errors correspond to smaller N. Indeed, we can observe that when N = 64, the errors are between orders of 10^{-12} and 10^{-15}. The conclusion we can get here is that, for bigger N, less error we get. This is not a surprise at all considering the results obtained in the previous section.

On the other hand, we will show the plots of the errors for the different derivatives and different values of N.

FIRST DERIVATIVE

We start the plots of the errors with N = 8.

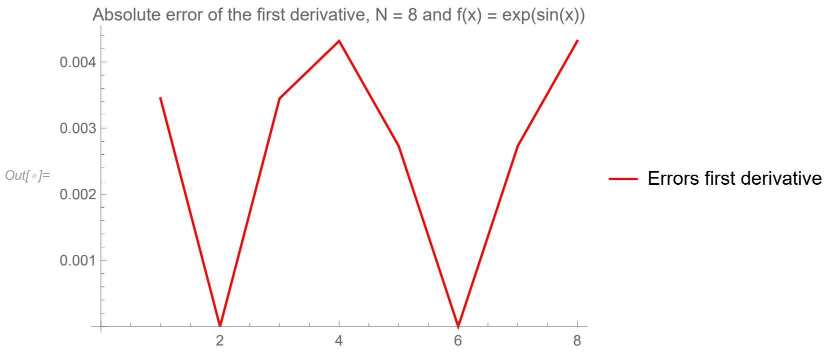


Figure 11: Plot of the error of the first derivative of $f_1(x)$ for N = 8.

We are showing now the one with N = 16.

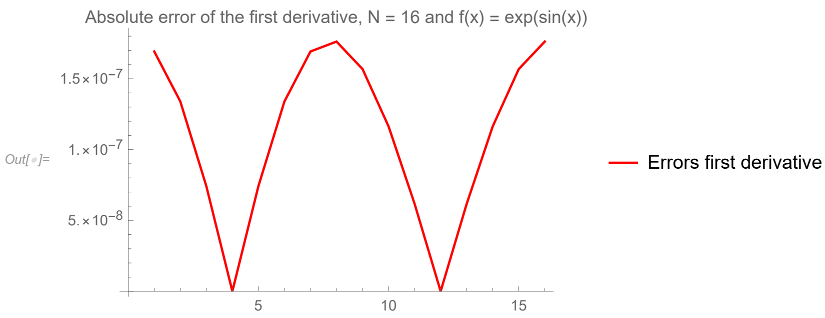


Figure 12: Plot of the error of the first derivative of $f_1(x)$ for N = 16.

We finally show the N = 64 plot.

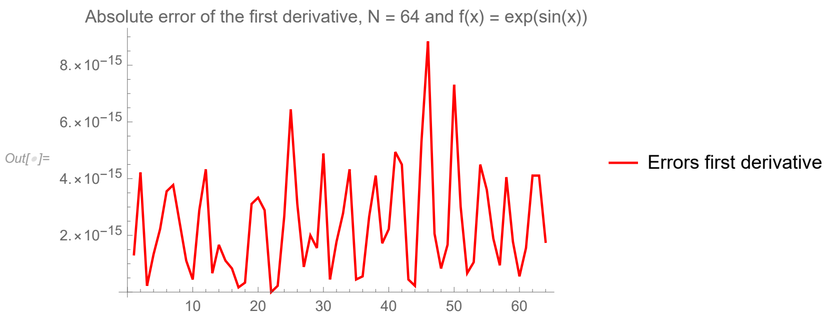


Figure 13: Plot of the error of the first derivative of $f_1(x)$ for N = 64.

As we have seen in the table before, errors for N = 64 are much smaller than the ones of N = 8, then the conclusion we get for the first derivative is the same we have got with the table. Bigger N, less error.

SECOND DERIVATIVE

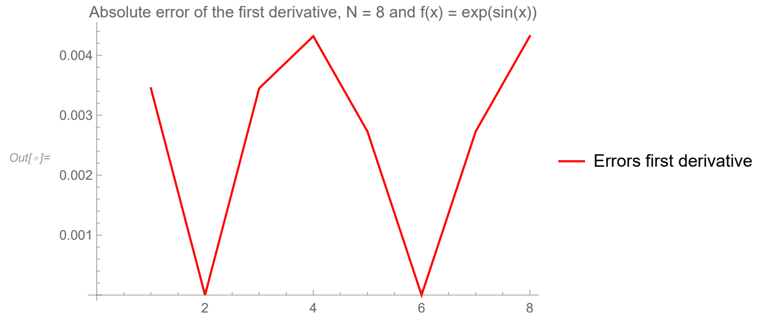


Figure 14: Plot of the error of the second derivative of $f_1(x)$ for $N = 8$.

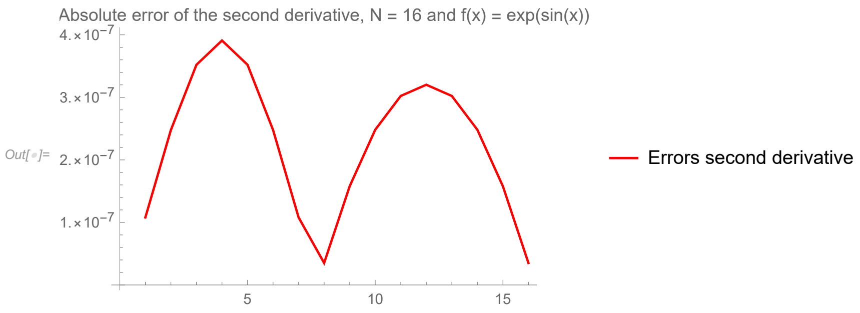


Figure 15: Plot of the error of the second derivative of $f_1(x)$ for $N = 16$.

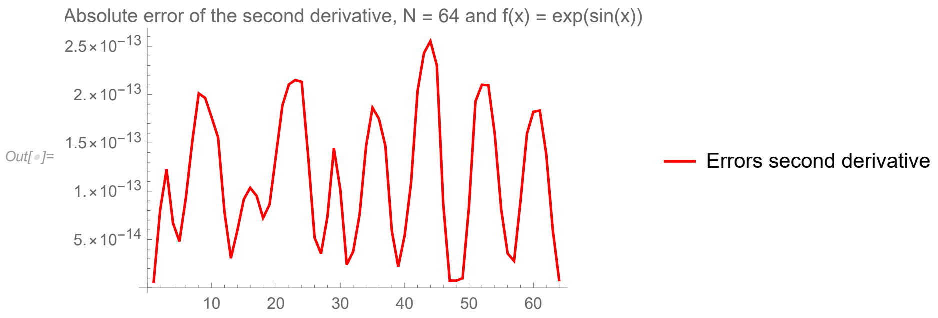


Figure 16: Plot of the error of the second derivative of $f_1(x)$ for $N = 64$.

The same happens for the second derivative, $N = 64$ errors are much smaller than the $N = 16$ and $N = 8$ ones.

THIRD DERIVATIVE

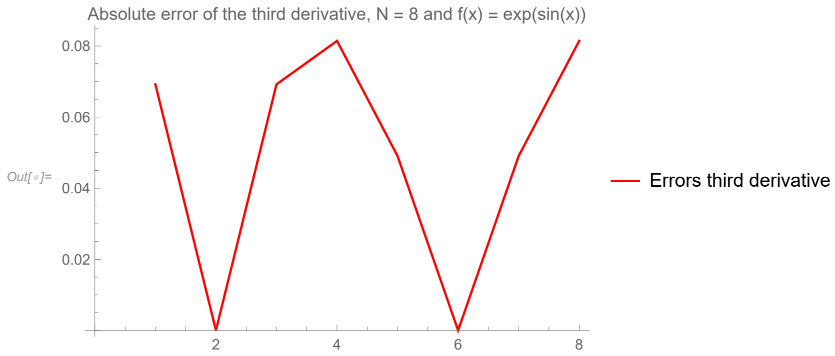


Figure 17: Plot of the error of the third derivative of $f_1(x)$ for $N = 8$.

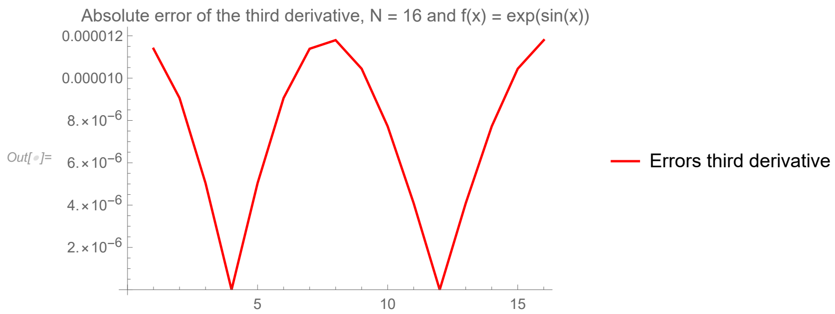


Figure 18: Plot of the error of the third derivative of $f_1(x)$ for $N = 16$.

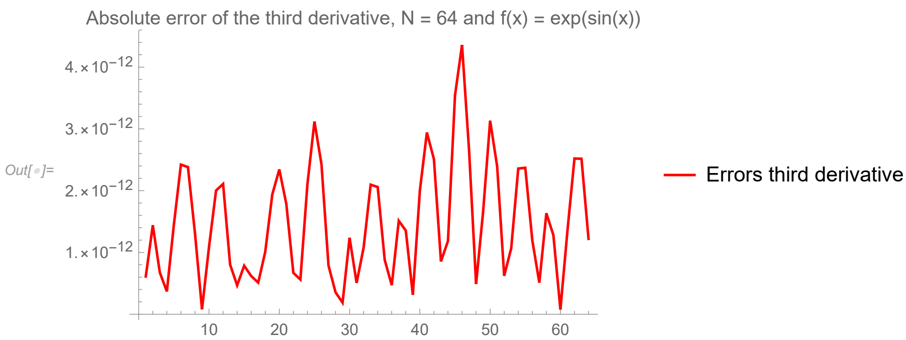


Figure 19: Plot of the error of the third derivative of $f_1(x)$ for $N = 64$.

Summing up, we can see that for each derivative the value of the error is smaller when the value of N is bigger, which corroborates what we have said before, that the approximation is better when the value of N is bigger.
We can also see that the error is higher when the order of the derivative is higher, that is, the error for the first derivative is smaller than for the second derivative and the error for the second derivative is smaller than the error for the third derivative.

2. function $f_2(x) = \exp\left(\sin(x)^3\right)$

In order to solve the problem for this new function, we will take the solution of the first function as example, and we will continue with the same structure used in 0.0.1.

Firstly we have plotted the original function and its derivatives using the '(prime) command, as we have done for the first function. Here is the resultant plot:

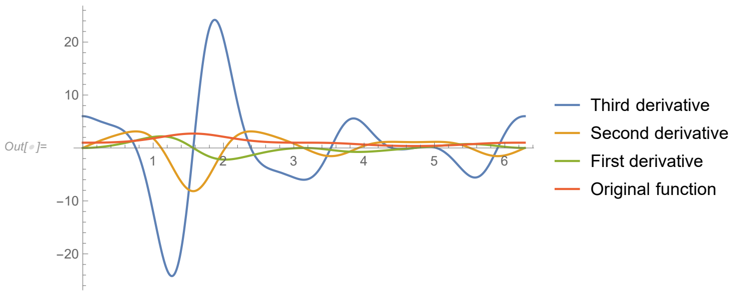


Figure 20: Plot of the original $f_2(x)$ function and its derivatives

Secondly, since we are asked to compare the exactly computed derivatives, which can be seen in the plot above, with the approximated ones using DFT, we will do it for the different values of N we have and then we will compare the results obtained.

FIRST DERIVATIVE

We start plotting the comparison between the exact computed first derivative with the approximated derivative for N =8.

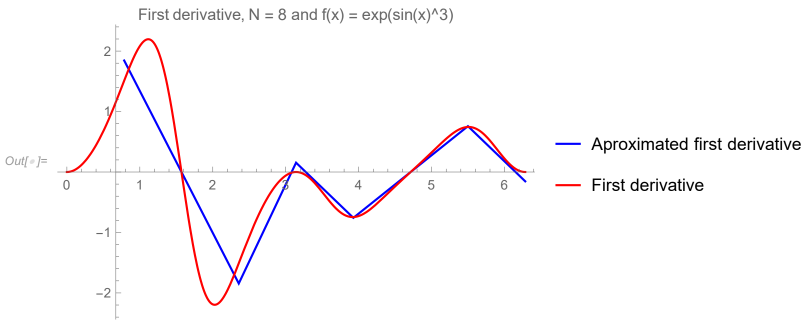


Figure 21: Plot of the comparison of the first derivative of $f_2(x)$ using DFT and Mathematica command ' (prime) for N = 8.

Once we do it for N = 8, we go with N = 16.

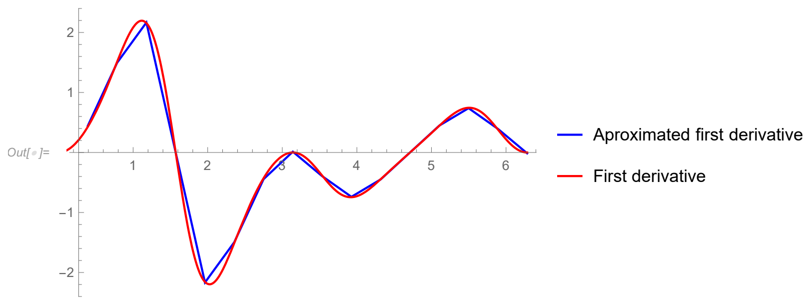


Figure 22: Plot of the comparison of the first derivative of $f_2(x)$ using DFT and Mathematica command ’ (prime) for N = 16.

We go with the last one, N = 64.

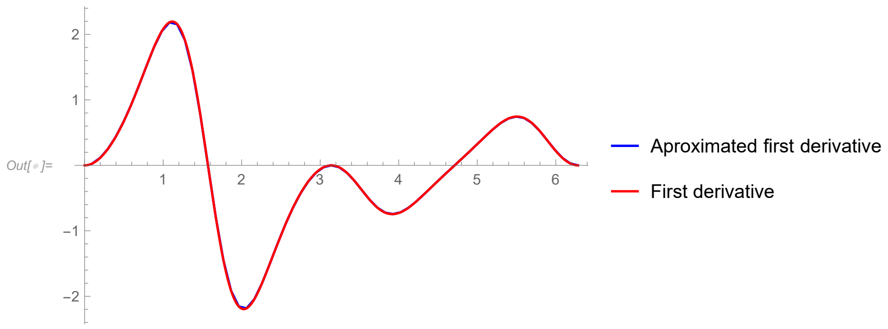
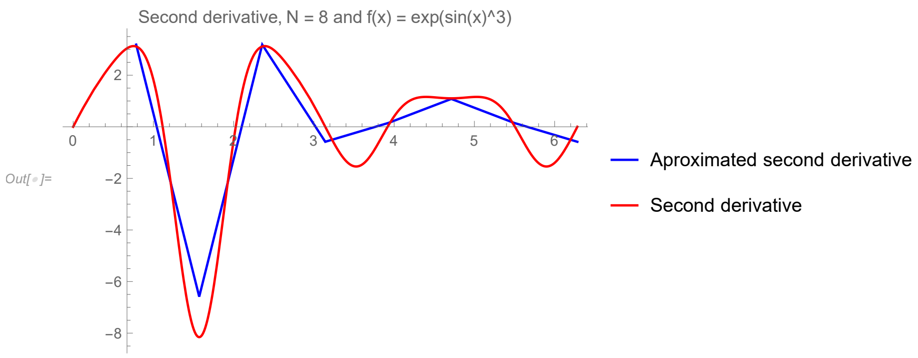


Figure 23: Plot of the comparison of the first derivative of $f_2(x)$ using DFT and Mathematica command ’ (prime) for N = 64.

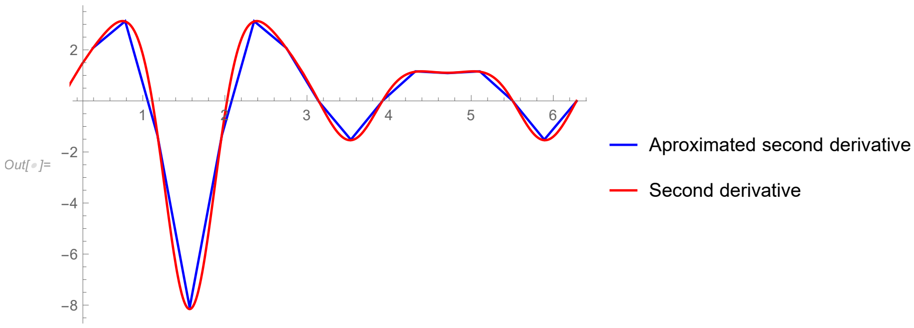
Taking into account the three plots above, we can see that as the value of N becomes bigger, the better the approximation becomes. This is not great news at this point due to is the same conclusion we have got for the different derivatives with $f_1(x)$. The approximation with N = 8 is very poor and we can see a big skip between N = 8 and N = 16 although the best approximation is achieved with N = 64.

<u>SECOND DERIVATIVE</u>

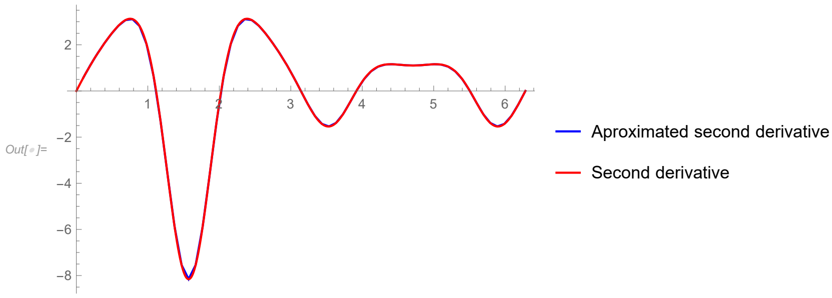
Let’s analyze now the second derivative, starting with N = 8.



We continue with N = 16.



Let's finish this second derivative analysis with N = 64.



This case is not different if we take into account previous conclusions, the best approximation is the one with N = 64. Although, in this case, we can appreciate that the approximation with N = 16 is a bit worse than the ones we have been analyzing when N = 16, in fact, the approximation is not just poor at the corners, is quite poor in all the plot. N = 8 does not surprises us, keeps being very poor.

<u>THIRD DERIVATIVE</u>

The last comparison between the exact derivative value and its approximated value we are doing is the analysis of this third derivative. We are starting with 8 number of points in the grid.

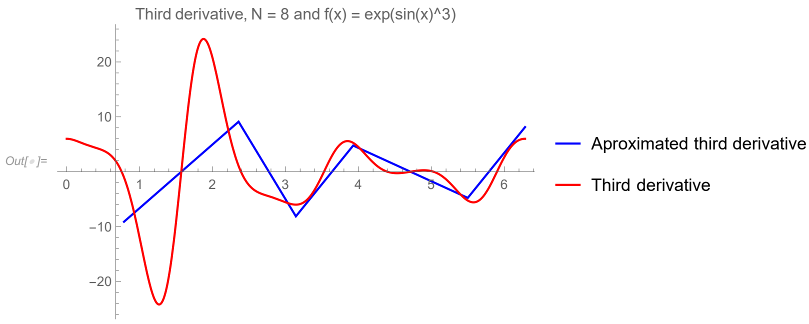


Figure 24: Plot of the comparison of the third derivative of $f_2(x)$ using DFT and Mathematica command ' (prime) for N = 8.

Next plot is N = 16.

The last plot is with N = 64.

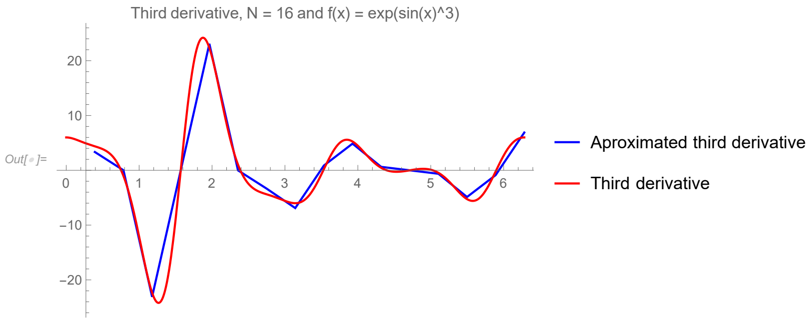


Figure 25: Plot of the comparison of the third derivative of $f_2(x)$ using DFT and Mathematica command ' (prime) for N = 16.

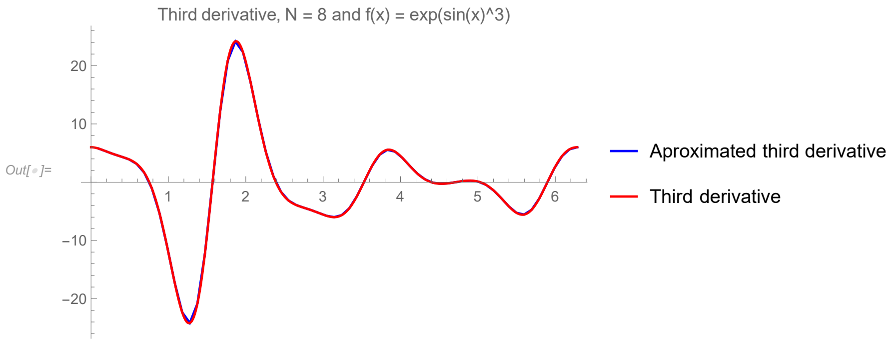


Figure 26: Plot of the comparison of the third derivative of $f_2(x)$ using DFT and Mathematica command ' (prime) for N = 64.

Here we can conclude the same thing as for the first and the second derivatives, that is, when the value of N is higher, the approximation is better.

We have to analyze also the absolute error of the computed derivatives. We will show a table where we put the absolute errors and we will also show the plots of the errors for different values of N.

On the one hand, in order to compare the absolute error, we have computed the absolute error of the computed derivatives and we put them in a table:

Values	N = 8	N = 16	N = 64
First derivative	0.333595	0.0154951	9.99201×10^{-15}
Second derivative	1.56751	0.0576027	2.02921×10^{-13}
Third derivative	8.06686	1.05324	4.32721×10^{-12}

As we have done for f_1, when analyzing the table of errors, we see that errors are relatively small for N = 64 compared to N = 8 and N = 16. Although there is no the difference it was for f_1 between N = 8 and N = 16 errors. This is, we can't deduce that N = 16 is going to be a good approximation due to its errors are not small enough.

On the other hand, we have the plots of the errors for the different derivatives and different values of N. Conclusions will be shown at the end of the plots of the three derivatives.

FIRST DERIVATIVE

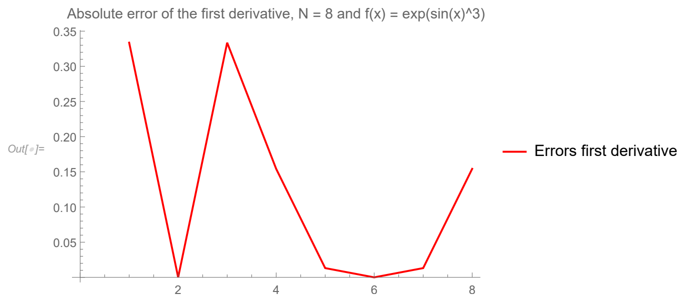


Figure 27: Plot of the error of the first derivative of $f_2(x)$ for N = 8.

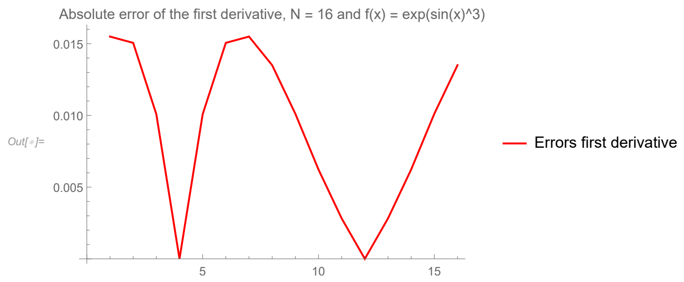


Figure 28: Plot of the error of the first derivative of $f_2(x)$ for N = 16.

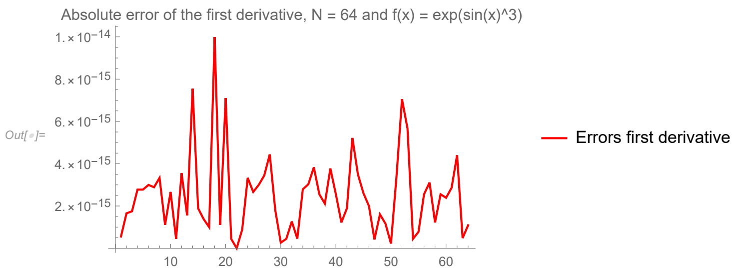


Figure 29: Plot of the error of the first derivative of $f_2(x)$ for N = 64.

SECOND DERIVATIVE

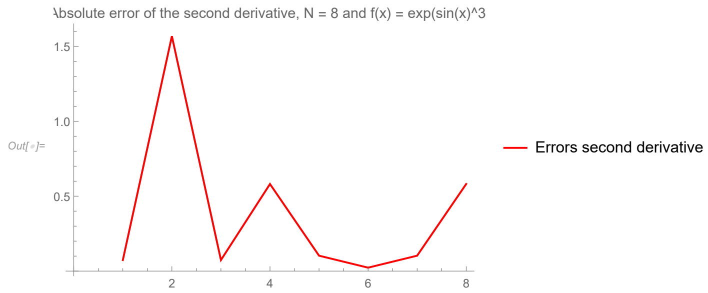


Figure 30: Plot of the error of the second derivative of $f_2(x)$ for $N = 8$.

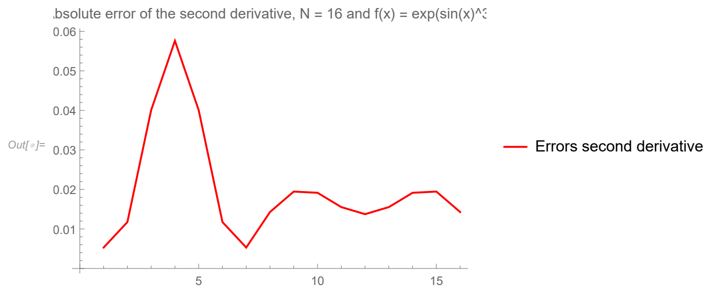


Figure 31: Plot of the error of the second derivative of $f_2(x)$ for $N = 16$.

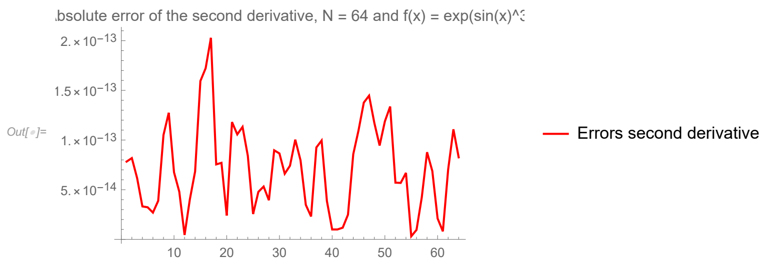


Figure 32: Plot of the error of the second derivative of $f_2(x)$ for $N = 64$.

THIRD DERIVATIVE

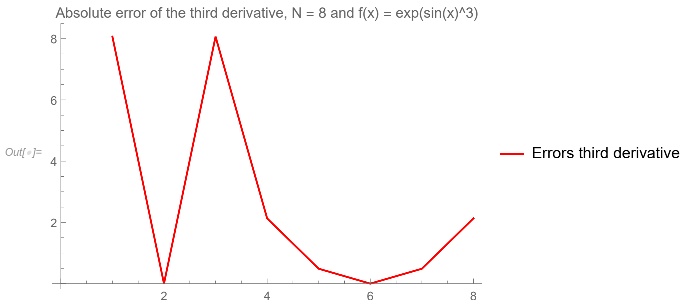


Figure 33: Plot of the error of the third derivative of $f_2(x)$ for N = 8.

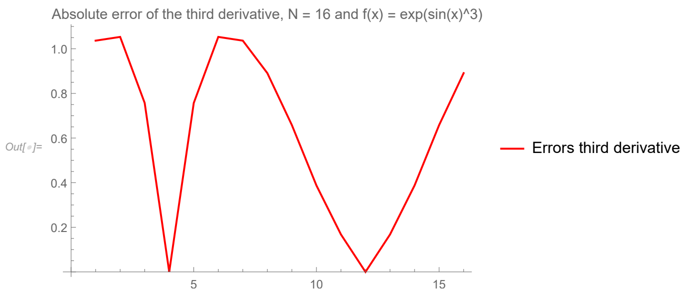


Figure 34: Plot of the error of the third derivative of $f_2(x)$ for N = 16.

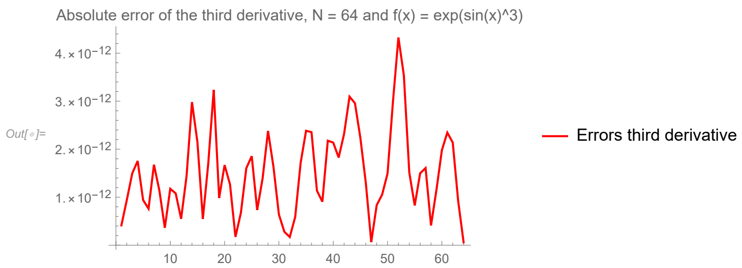


Figure 35: Plot of the error of the third derivative of $f_2(x)$ for N = 64.

As we have seen before, the approximation is better when the value of N is bigger and talking about the error, this translates to the error being smaller when the value of N is bigger, which can be seen in the plots above.

Respect to the order of the derivatives, we can see that the error is bigger when the order of the derivative is higher, that way, the biggest error corresponds to the third derivative and the smallest one corresponds to the first derivative.

4.2 Spectral derivatives

As already said during the theoretical classes and in the computer programming on the Discrete Fourier Transform, we can use the DFT in order to **approximate derivatives** of a smooth, periodic function $v(x)$. The resulting derivatives approximations are called **spectral derivatives**. They can be extremely accurate if v is very smooth in addition to being periodic.

4.2.1 The variable coefficient wave equation

Statement:

$$u_t + c(x)\, u_x = 0, \qquad c(x) = \frac{1}{5} + \sin^2(x-1), \qquad x \in [0, 2\pi], t > 0, \tag{2}$$

with initial condition

$$u(0,x) = \exp[-100(x-1)^2].$$

This function is not mathematically periodic, but it is so close to zero at the ends of the interval that it can be regards as periodic in practice.

In the reference Lloyd N. Trefethen, *Spectral Methods in Matlab*, Program 6 on page 26 solves this matter for the variable coefficient wave equation.

(a) Write a program in *Mathematica* that, in order to solve (2) and to reproduce the 3D plot seen in the theoretical class, follows the steps:

- For the time derivative you use a leap-frog formula

$$u_t = \frac{v_j^{n+1} - v_j^{n-1}}{2\,\Delta t}$$

- You approximate the spatial derivative spectrally.

The approximation becomes

$$\frac{v_j^{n+1} - v_j^{n-1}}{2\,\Delta t} = -c(x_j)\,(Dv^n)_j, \quad j = 1, \dots, N.$$

Use the function

$$u(t,x) = \exp[-100(x - 1 - c(x)\,t)^2]$$

in order to define the two initial approximations needed in order to proceed.

Consider the case of $N = 128$ and $\Delta t = 10^{-2}$.

Plot the 3D graphic of the solution for $0 \le x \le 2\,\pi$ and $0 \le t \le 8$.

Solution:

The first thing we have to do is to represent the two initial approximations. Then, let's do it and let's compare them with the two original ones:

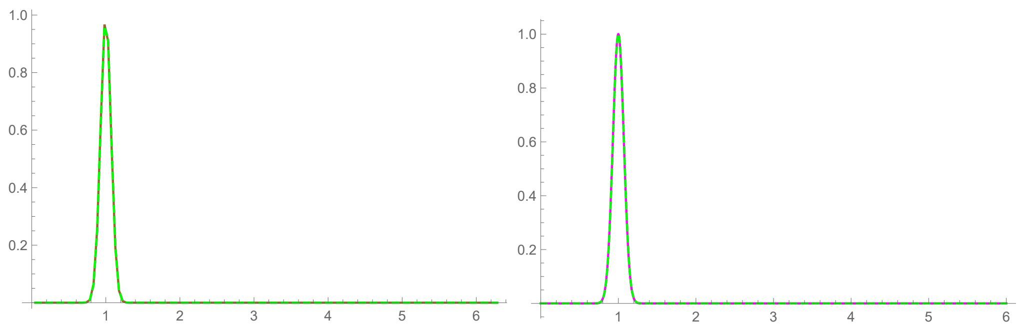


Figure 36: The two initial approximations(Green) & The two original approximations (Pink)

As we can see, the approximations are pretty good. They are a little bit different at the top corner, but this can be improved choosing a bigger N, instead of the expression N = 128 that we are using in this example.

Now, let's plot the solution for different values of t ($0 \leq t \leq 8$). We are choosing a step of 0.4, resulting 21 plots in total:

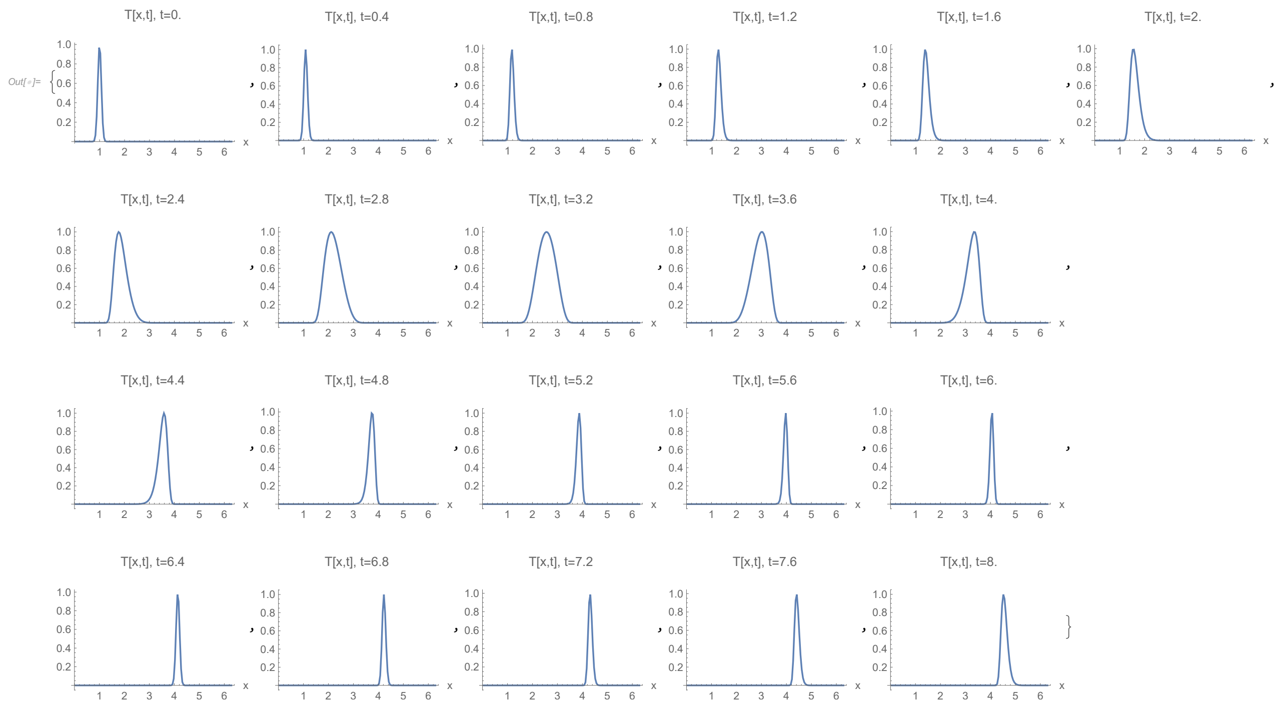


Figure 37: Solution for different values of t

We notice that the wave goes from left to right. It starts slowly, but then it speeds up until it reaches its maximum velocity, around t = 4. Then, it starts slowing down again.

The corresponding 3D plot is the following one:

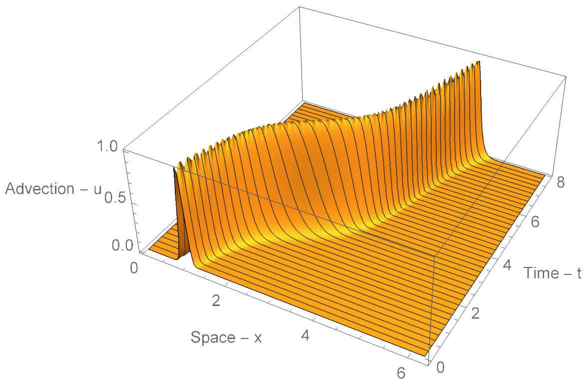


Figure 38: 3D plot of the solution

It is clear, that the 3D plot corresponds to the wave we have seen at Figure 37. Furthermore, we conclude the same as in Figure 37: the wave goes from left to right and its velocity increases until around t = 4.