Signal and Image Processing

Lab session 2

Dirac comb, short-term spectrum

Elliott Perryman, Jan Zavadil

8.10.2021

1 Excercise 1: *Construction of the Dirac comb*

1.1 Write a program to plot the N first terms of the series: $1 + 2\cos(2\pi t) + 2\cos(4\pi t) + \ldots + 2\cos(2\pi N t) + \ldots$

The code to compute this sum is in Appendix A. This code simplifies the code to:

$$S_n = 1 + 2\sum_{n=1}^{N} cos(2\pi n t) \quad (1)$$

and works for scalars and vectors t, since numpy's cosine function is vectorized. The application of this sum for $N = 10^3$ over the domain [0,10] is shown in Figure 1, where the spacing of t is determined by N. Specifically, $t[i+1] - t[i] = \frac{1}{10N7}$.

1.2 What happens when N tends to infinity $+\infty$?

We receive a function converging to Dirac comb Δ_1. That is

$$\Delta_1 = \sum_{n\in\mathbb{Z}} \delta_n = \begin{cases} 0 & \forall x \in \mathbb{R} \setminus \mathbb{Z} \\ +\infty & \forall x \in \mathbb{Z} \end{cases} \quad (2)$$

1.3 Does this series converge point-wise?

No, for a series $A_n = \sum_{n=0}^{+\infty} a_n$ to converge, it is necessary, that $a_n \to 0$, which is not true in this case.

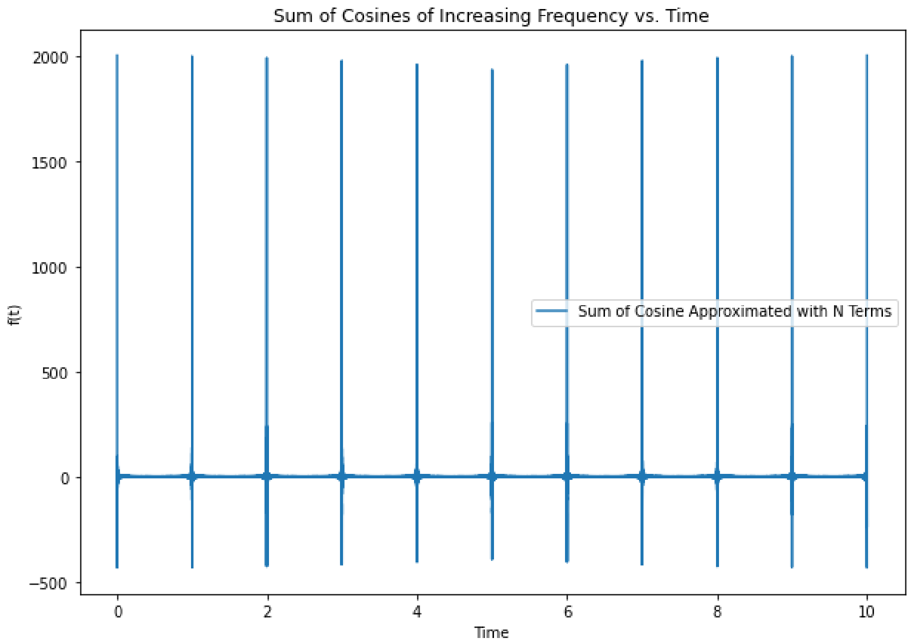

Figure 1: Sum of Cosines of Increasing Frequency vs. Time. The sampling rate of the time is chosen to be $N107$ so that there are points with spacing within the period of the highest frequency component. As the number of terms increases, the sum of functions converges to the Dirac comb

1.4 Explain why the series could be written $\sum_{n\in\mathbb{Z}} e^{2i\pi nt}$

$$\sum_{n\in\mathbb{Z}} e^{2i\pi nt} = 1 + \sum_{n=1}^{+\infty} e^{2i\pi nt} + e^{-2i\pi nt} = 1 + \sum_{n=1}^{+\infty} 2\cos(2\pi nt) \tag{3}$$

1.5 Denoting by $\Delta_a = \sum_{n\in\mathbb{Z}} \delta_n a$ and assuming that $\hat{\Delta}_a = \frac{1}{a}\Delta_{\frac{1}{a}}$, show that the series is actually the Dirac comb.

Applying the Fourier Transform on δ gives a sum of exponentials over $\mathcal{Z}$ which is the alternate representation derived in 1.4.
In our case $a = 1$,

$$\hat{\Delta}_1 = \sum_{n\in\mathbb{Z}} \hat{\delta}_n \stackrel{F}{=} \sum_{n\in\mathbb{Z}} e^{2i\pi nt}. \tag{4}$$

1.6 How many points between 0 and 1 do we have to choose to plot correctly the last term in the truncated series? Check that this constraint is well fulfilled.

The number of points needed depends on the frequency of the last term, which is given by N. Period of the last term $\cos(2\pi Nt)$ is $\frac{1}{N}$, we use in total $N7$ points between 0 and 1, which leaves 7 points for a single period of the last cosine function.

1.7 What happens if we consider the limit of $1 + 2\cos\left(\frac{2\pi t}{Q}\right) + 2\cos\left(\frac{4\pi t}{Q}\right) + \ldots + 2\cos\left(\frac{2\pi Nt}{Q}\right) + \ldots$ Give a theoretical argument to justify this behaviour. Write the corresponding program.

The series no longer converges to Δ_1 but to Δ_Q. This is caused by a change of periodicity of cosine functions in the sum, by changing Q we can set the width between two peaks of the Dirac comb.
The code to compute this sum is in Appendix B. Its output is shown below in Figure 2. The parameter N is set to 10^3, with the number of points in t set to $\frac{10N7}{Q}$, giving the same 7 points over a single period of the highest frequency in case $Q = 1$. The value of Q=1,2,4 is shown, with the width of the comb corresponding to the value of Q as expected.

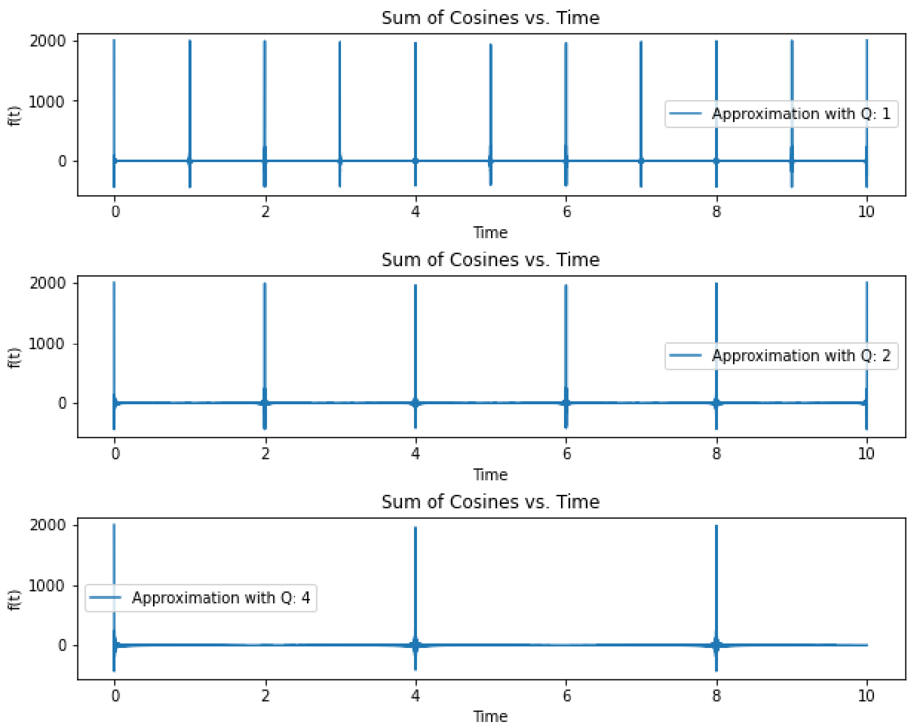

Figure 2: Sum of Cosines of Increasing Frequency with Divisor Q vs. Time. The sampling rate of the time is chosen to be $\frac{10N7}{Q}$. As the number of terms increases, the sum of functions converges to the Dirac comb with width Q, where the points x=nQ, with n an integer, go to infinity and all other points going to 0.

2 Excercise 2: *Short-term Fourier transform*

2.1 Plot the modulus of DFT of specified signal. You observe a symmetry in the representation, does that seem correct? What is the reason for such behaviour?

The modulus of the DFT is shown below in Figure 3. In this figure, the output is symmetric around the frequency $f = 0$. A single periodic signal with frequency would have a peak at $\pm k$, corresponding to its representation as $2 * (e^{2\pi ikt} - e^{-2\pi ikt})$. For the input signal, there are two frequencies contained, so there are peaks at $\pm k_1$ and $\pm k_2$.

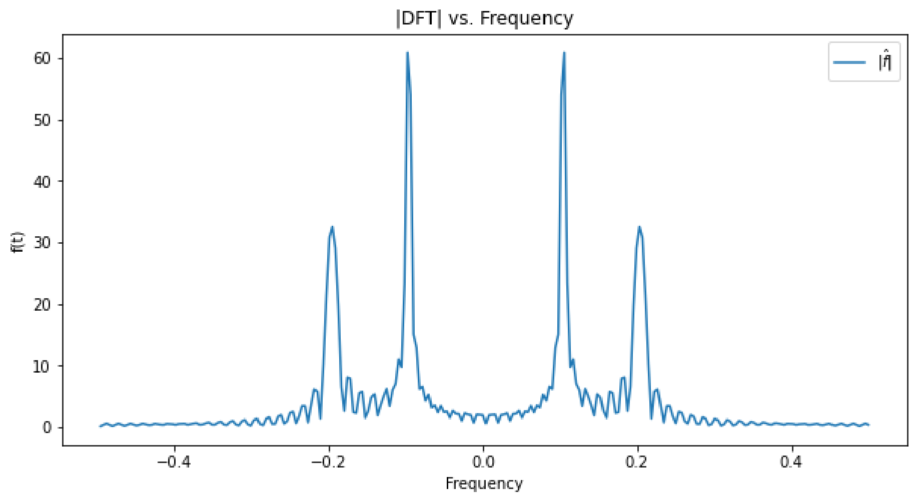

Figure 3: Modulus of DFT vs. Frequency.

2.2 If you add zeros to this initial sequence to obtain a sequence of size 512, what is the impact on frequency resolution. Give a detailed explanation.

Figure 4 shows the modulus of the Fourier transform of this signal. Comparing this with Figure 3, they appear to have essentially the same resolution. In our opinion, this makes sense because the zero-padding does not add any oscillations or terms to the Fourier representation of the signal. A nonlinear least squares fit to a Lorentzian around the -0.1 frequency peak gives integrated widths of $\beta = 0.0130$ for the shorter signal and $\beta = 0.0135$ for the longer signal, which seems to indicate that perhaps the shorter signal has higher resolution, but both resolutions are similar.

The only visible difference between the two signal spectres is in the frequencies around the two peaks where we can see that the longer signal gives a spectrum with sharper and narrower ghost frequencies, which could mean better resolution in this case.

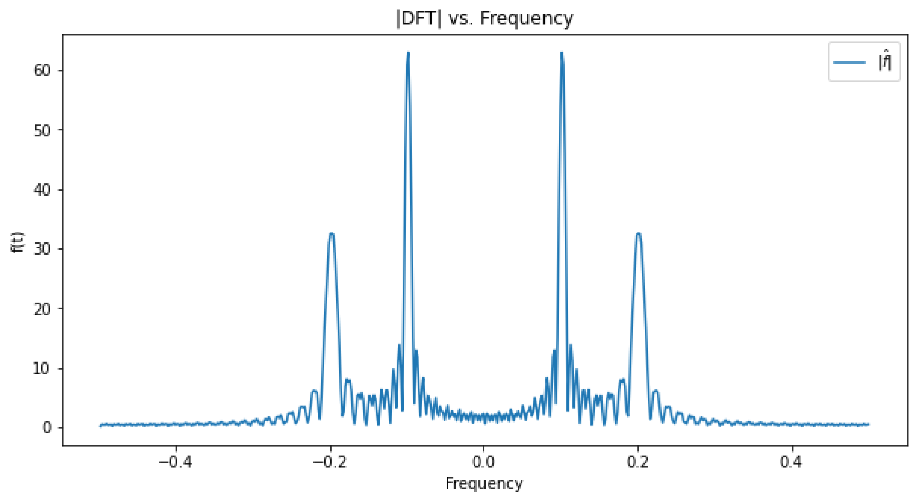

Figure 4: Modulus of DFT vs. Frequency with 512 points.

2.3 Create a time-frequency mesh according to specifications. Plot the modulus of the final matrix using contour command.

This is shown in Figure 5. Initially, the window covers the time from t=0 to t=64, so the Fourier transform of this has a peak frequency at 0.1 and -0.1. As the window moves through the signal, it reaches a point where both the 0.1 and 0.2 Hz signals are in window. This happens when the center time (y axis) is about 128. Eventually, the only signal is the 0.2 Hz sine curve, giving a peak at ± 0.2. The signal eventually goes to zero. The window size in this figure is 64.

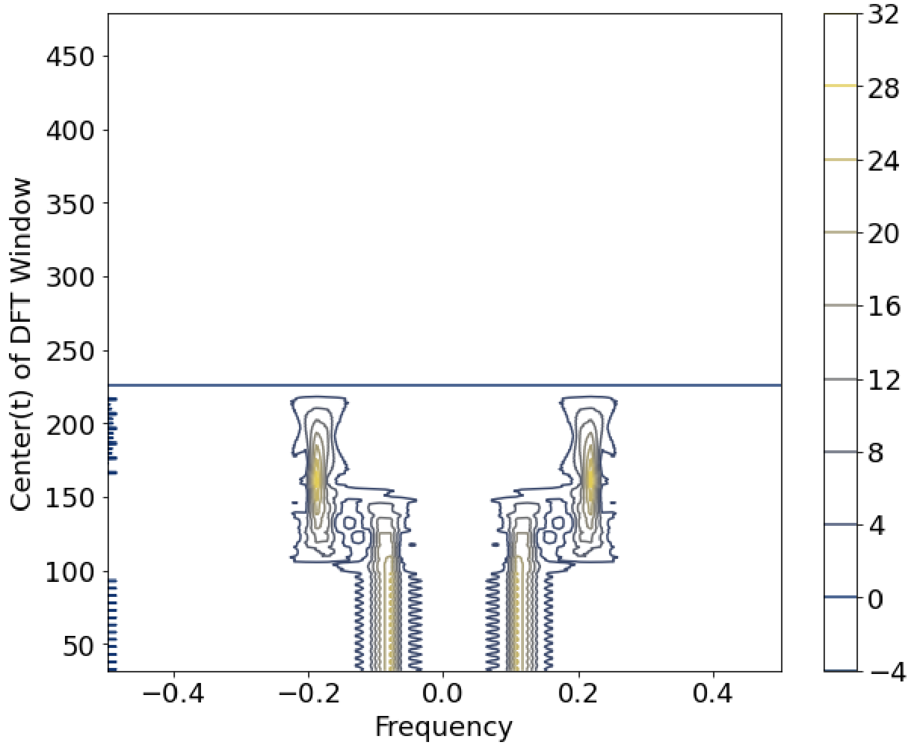

Figure 5: Center of DFT Window(t) vs. Frequency. The z coordinates correspond to the modulus of the DFT of the signal. To stretch out the signal to be visually, the image has been linearly interpolated.

3 Appendices

3.1 Appendix A. Sum of Cosines

Listing 1: Evaluate Sum of Cosines at Time t

```
import numpy as np
def cos_sum(N, t):
    return 1 + 2*np.sum([np.cos(2*np.pi*t*i) for i in range(1,N+1)], 0)
```

3.2 Appendix B. Sum of Cosines with Divisor Q

Listing 2: Evaluate Sum of Cosines at Time t with Divisor Q

```
import numpy as np
def cos_sum(N, t, Q):
    return 1 + 2*np.sum([np.cos(2*np.pi*t*i/Q) for i in range(1,N+1)], 0)
```