Introduction

Sinusoidal functions are very prevalent in everyday life, but many times they are quite subtle that our awareness of its applications seem to become infrequent. But they are everywhere in our daily life, and it's important that we know how to model them.

Section 1. The raw data

The graph above describes the intensity and pitch of sound within a miniscule duration of a chime playing. This data was generated on https://academo.org/articles/oscilloscope/. In this case, the intensity or amplitude is mapped onto the y axis, with arbitrary values, with the implication that absolute intensity is not measured in this case, rather the relative intensity is. The x axis represents time in milliseconds (ms), and the pitch is inversely proportional to the wavelength.

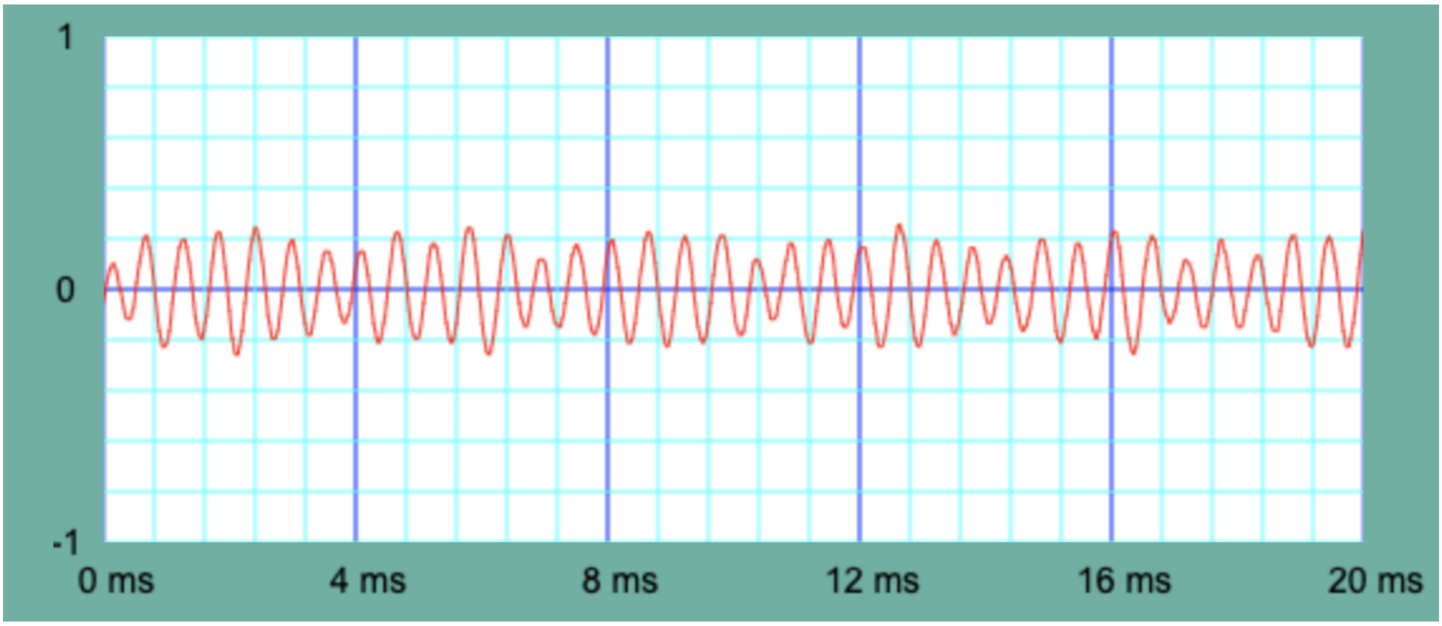


Figure 1. Oscilloscope data of a chime playing, taken from ("Oscilloscope").

In the situation with the software, it didn't provide any further information in the form of numerical or equations, but rather only a simple visual representation of the data, this graph. Therefore in order to perform the analysis that follows, I had to define 14 data points myself based on the snapshot of the oscillator. This of course leads to an inaccuracy prone data set and analysis which will slightly affect the reliability, discussed towards the end of this paper.

Section 2. Data Modification and Selection

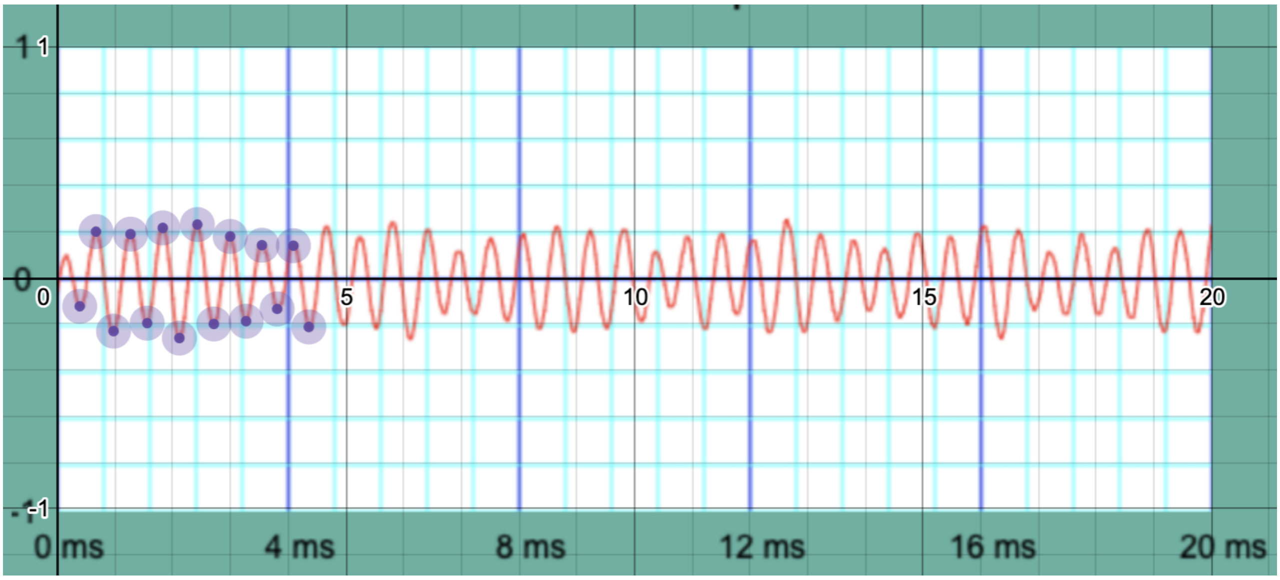


Figure 2. Modified oscilloscope data, conducted on (“Desmos”)

In order to identify and define my 14 different data points, I found that the most convenient way to do so was on Desmos. Since the axises were scaled differently from each other, I had to perform additional modification of the image above to line up the x and y axis values between the image layer and the desmos layer, in addition to moving the image around to line up the origin as shown in figure 2.

Afterwards, I began to place the 14 different points in a way that wasn’t so bunched up or spread out that the soon to be performed regression model could potentially be completely inaccurate or unrepresentative of the overall structure of the data. An example of these are shown in figure 3 and 4 below. Looking at figure 2, it is evident that the data points are sufficiently spaced out and balanced, where 14 points cover 7 cycles. By having the data points defined at the troughs, or the maximums and minimums of the wave, it also maintained the overall composition of the wave, as its structure is consistent throughout.

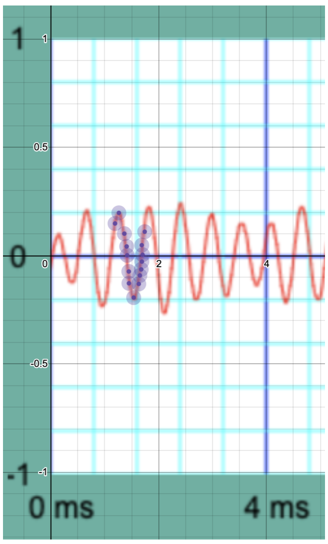


Figure 3. Data points are too bunched up, only covering 1 cycle

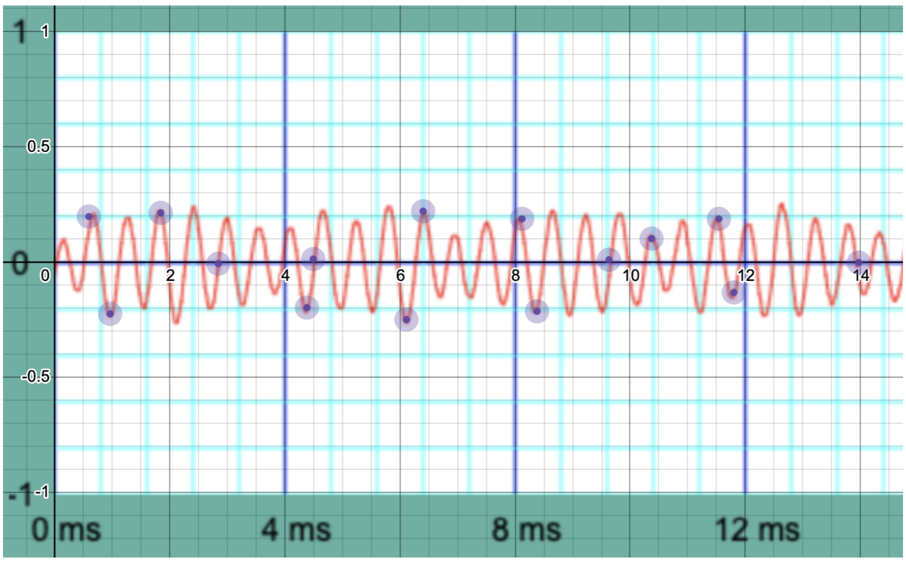


Figure 4. Data points are too spaced out, with 14 data points covering over 25 cycles

Figure 2 thereby results in the following data points and coordinates:

time (ms)	intensity	corresponding coordinates
0.373	-0.120	(0.373, -0.120)
0.655	0.204	(0.655, 0.204)
0.959	-0.227	(0.959, -0.227)
1.253	0.193	(1.253, 0.193)
1.542	-0.192	(1.542, -0.192)
1.812	0.220	(1.812, 0.22)
2.100	-0.257	(2.100, -0.257)
2.411	0.234	(2.41, 0.234)
2.697	-0.196	(2.697, -0.196)
2.980	0.183	(2.980, 0.183)
3.255	-0.184	(3.255, -0.184)
3.530	0.145	(3.530, 0.145)
3.794	-0.131	(3.794, -0.131)
4.075	0.143	(4.075, 0.143)
4.342	-0.209	(4.342, -0.209)

Figure 5. Data points and corresponding coordinates of defined dataset
*rounded to 3 decimal points (3-4 significant figures)

As obviously apparent in both figure 2 and figure 5, intensity values of the minimums and maximums do not stay constant throughout, with a standard deviation of 0.0459 and 0.0350 respectively (figure 7). There is also a very slight difference between the max and min values, differing at 0.001 as seen in figure 7. Less noticeably, the differences between times at which the minimums and maximums occur are also not consistent, albeit the differences are very miniscule with a standard deviation of 0.0134 (figure 7). These inconsistencies make sense given the context that the oscilloscope wasn't detecting perfect pitched sounds. Although they are not the most ideal to fit a perfect sinusoidal function onto, it can still be worked with.

Figure 6. differences between each consecutive time values (ms)

time (ms)	difference between each consecutive time values (ms)
0.373	
0.655	0.282
0.959	0.304
1.253	0.294
1.542	0.289
1.812	0.27
2.1	0.288
2.411	0.311
2.697	0.286
2.98	0.283
3.255	0.275
3.53	0.275
3.794	0.264
4.075	0.281
4.342	0.267

	mean	standard deviation
minimum values of intensity	-0.190	0.0459
maximum values of intensity	0.189	0.0350
difference between time values of mins and maxs	0.284	0.0134

Figure 7. Mean and standard deviation of the intensity and time differences of the data points as calculated by the AVERAGE and STDEV functions on google sheets

Section 3. Parameters

The values investigated above can be used to determine the following parameters that are involved in the data set: the minimum, maximum, range and period.

The minimums and maximums of this data lies in the ranges of $-\ 0.257 \le y \le -\ 0.120$ and $0.143 \le y \le 0.234$ respectively. However, I had to define a singular value that can describe these ranges, in order to manually calculate a model. I looked at the analysis in figure 7 above, which found the mean of those varying minimum and maximum values to be -0.190 and 0.189 respectively. They were a decent representation of the values and also corresponded with each other very closely, and were therefore set to be the minimum and maximum values.

Likewise, the range can be defined two different ways. One treats the dataset without the underlying intent to fit it in regular sinusoidal function, which would give the range of $-\ 0.257 \le y \le\ -\ 0.234$(the lowest within the range of minimum values and the highest within the range of maximum values. The other definition will treat it with the intention of calculating a regular sinusoidal representation, which can be derived from the average minimum values and maximum values earlier, resulting a range of $-\ 0.190\ <\ y\ <-\ 0.189$.

For this particular data type, there is a constraint that the domain must comply with, in that there can be no negative values as we are working with time. In most frames of reference, time must be a positive value, therefore the domain is all x in real numbers, for x larger or equal to 0.

Lastly, the period can be calculated by observing the values in figure 6 and the analysis in figure 7. The values listed on the second column of figure 6 essentially describe each half period within the data set, as they are the difference between the x axis values of each maximum and minimum (½ a period) as opposed to each maximum and the next or each minimum and the next (whole period). Thus, by taking the average of those differences and multiplying it by 2, it will result in the period, which in this case is 0.284*2 = 0.568ms

Overall this data can be summarized into the following:

minimum	y = -0.190
maximum	y = 0.189
range	*{y | y∈R, 50≥y≥87}*
domain	*{x | x∈R, 0≤y}*
period	0.568ms

Section 4. Model M - analytical model

In order to describe the oscilloscope data, a sinusoidal function is likely most fit as it has been implied a few times previously. Thus, in order to best represent the overarching data, I decided proceed using a sine function,

$$M = Asin(B(x + C)) + D$$

where A describes its vertical dilation, B describes its horizontal dilation, C represents the true horizontal shift, and lastly D is the vertical shift. This decision was due to the fact that the wave starts close to the centerline at x=0, as well as the other 4ms intervals as seen in the visual representation of the data (figures 1, 2). Only towards larger x values does the maximum points start to be horizontally closer.

Given the contextual understanding that the centerpoint of soundwaves as detected by oscilloscopes are always at y=0 regardless of what the minimum and maximum values are. As a result, the D value or the vertical shift of the analytical model will have to be 0. The rest of the equation will have to be determined with this established principle in mind. Thus a modified cosine function will be used in replacement,

$$M = Asin(B(x + C))$$

The **vertical dilation** can be calculated by dividing the difference of the maximum and the minimum and dividing it by 2, $\frac{0.189-(-0.190)}{2}$= 0.1895. I determined that the vertical dilation will remain positive as like a positive sine wave, it starts from 0 and goes upwards, not downwards, otherwise it would be negative.

∴ A = 0.1895

Technically, this does imply that the D value should be -0.0005 as it is the midpoint between the average minimum and maximum values, $\frac{0.189-0.190}{2} = -0.0005$. However, due to the fact that a miniscule difference of 0.0005 of vertical shift to the overall dataset is negligible, and more importantly, the previous establishment that D has to be 0, the implication of D=-0.0005 can be dismissed.

The **horizontal dilation** can be found through the following formula, $B = \frac{2\pi}{P}$, where P is the period. As I have calculated in Section 3, the period of this function is 0.568. Therefore the B value is equivalent to $\frac{2\pi}{0.568}$ or $\frac{\pi}{0.284}$. At this stage, I refrained from converting this irrational fraction into as it can lose a certain extent of accuracy, thus proceeding with a fraction instead.

∴ B = $\frac{\pi}{0.284}$

So far this is how the analytical model stands, with one parameter left to identify,

$$M = 0.1895sin(\frac{\pi}{0.284}(x + C))$$

Initially I implemented the following method to find C, which includes isolating the C variable and attempting to find the average when all 15 points were used as input. However, I quickly realized that this method isn't a sensible option. My decision to use the average minimum and maximum value instead of its absolute counterparts meant that I excluded several data points out of the domain of the function. Therefore, when inputting these invalid values into the code I wrote, it came out as undefined.

$$M = A\sin(B(x + C))$$

$$\frac{M}{A} = \sin(B(x + C))$$

$$\sin^{-1}\left(\frac{M}{A}\right) = B(x + C)$$

$$\frac{\sin^{-1}\left(\frac{M}{A}\right)}{B} = (x + C)$$

$$C = \frac{\sin^{-1}\left(\frac{M}{A}\right)}{B} - x$$

$$C = \frac{\sin^{-1}\left(\frac{M}{0.1895}\right)}{\left(\frac{\pi}{0.284}\right)} - x$$

$$\therefore C = \frac{0.284\sin^{-1}\left(\frac{M}{0.1895}\right)}{\pi} - x$$

As a result, I had to resort to only performing the same process but only for one point instead. The point chosen has to be within the domain and situated as close to its respective minimum or maximum as possible, as well as the sine wave itself. At this stage, two points would be satisfactory, (2.98,0.183) and (3.255, -0.184), which I will denote as points P and O respectively as shown in figure 8.

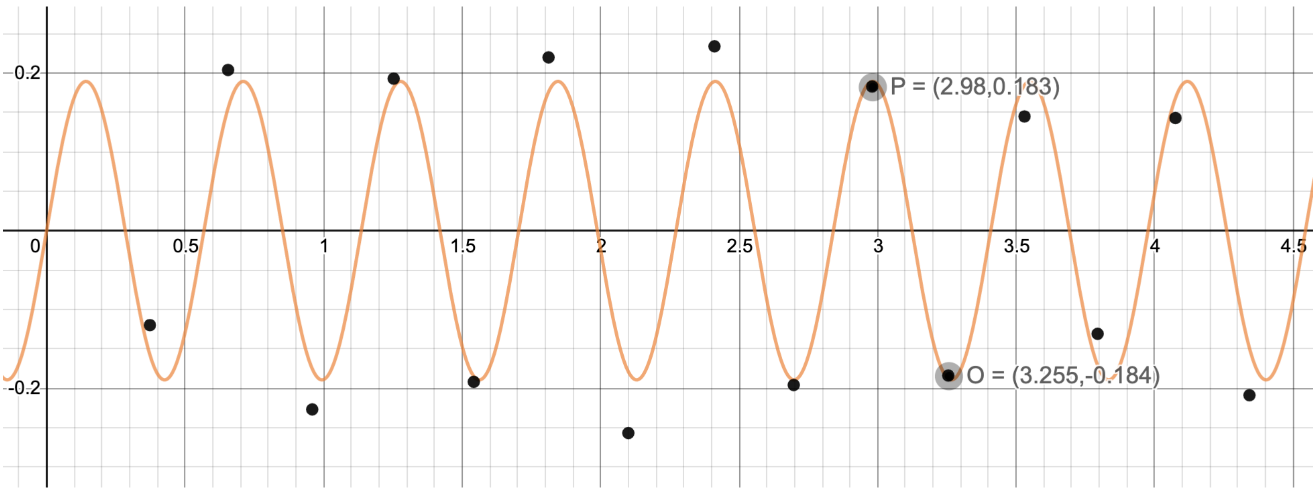


Figure 8. Points P and O are highlighted as the least distance point that is still within the domain.

In order to narrow down to one point, I checked for the one with minimal residual to find the point the function will best fit. If we were to compare figures 9 and 10, it is apparent that point O will be closest fit to the line as it is only 0.011 off. Therefore, I decided to use this data point to plug into the equation to find the C variable.

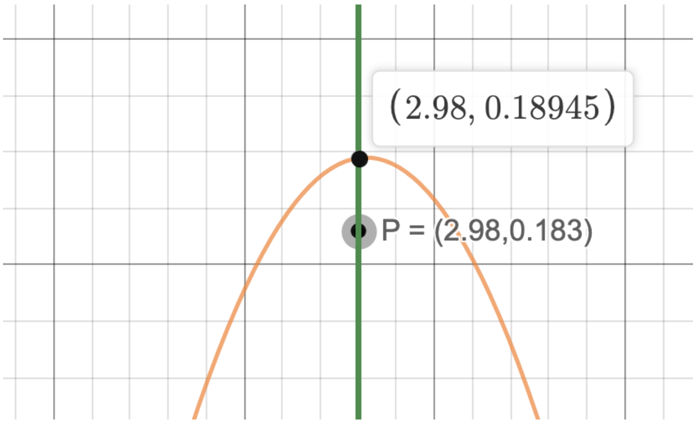


Figure 9. Point P, has a difference of 0.00645

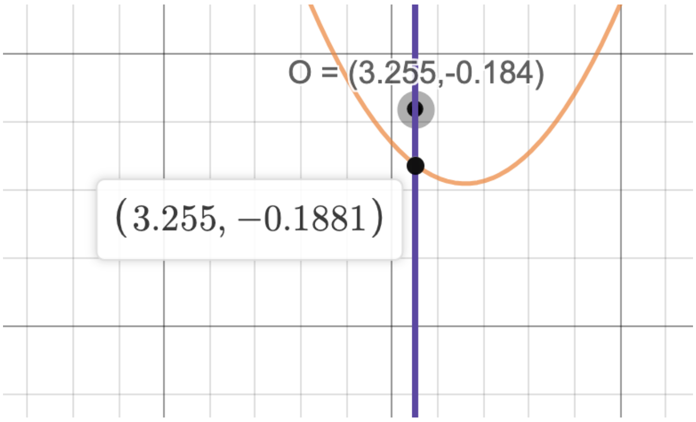


Figure 10. Point O has a different of 0.0041

The following is the result of plugging in x=3.255 and y=-0.184 into the equation and performing the calculation in desmos,

$$C = \frac{0.284 sin^{-1}(\frac{M}{0.1895})}{\pi} - x$$

$$C = \frac{0.284 sin^{-1}(\frac{-0.184}{0.1895})}{\pi} - 3.255$$

$$C =- 3.37516685789$$

$$\therefore C =- 3.375$$

However, I quickly realized that the C value can easily be simplified further simply by taking -3.37516685789mod0.568. By taking the modulus of the current C value against the period of the data, I am essentially eliminating the unnecessary shift opposed when inputting a data point that is much further away from x=0. This modulo operation results in 0.0328331421098. In order to verify that this has resulted in a similar sine function as the original C value did, I plotted two functions with the different C values as seen in figure 11.

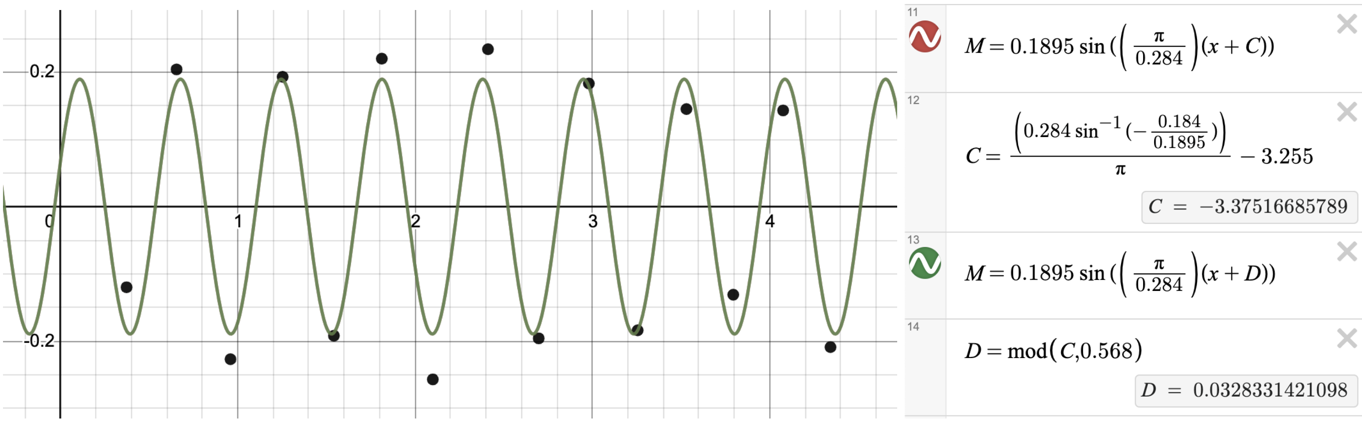
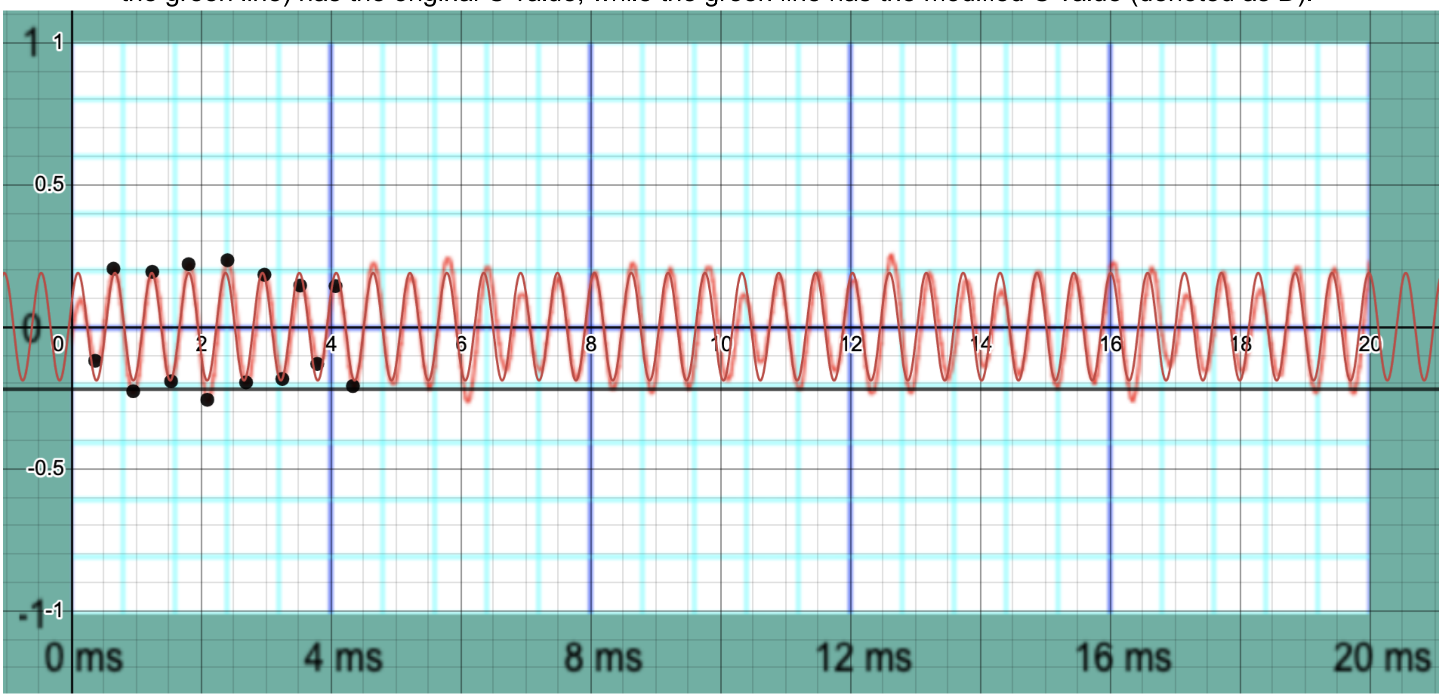


Figure 11. Verification of modified C value, where the red line (not visible as it was directly beneath the green line) has the original C value, while the green line has the modified C value (denoted as D).



As the figure suggests, these two C values produce similar sine functions, with the modified C value being more simplified. Therefore I decided to proceed with a modified C value of 0.0328331421098. This function has a domain of (0, 20) so it only accurately represents values within the image given but can not extrapolate as the behaviour of this sound is very uncertain, and has a maximum of 0.1985 and a minimum of -0.1985.

$$\therefore M = 0.1895 sin(\frac{\pi}{0.284}(x + 0.0328331421098))$$

Section 5. Model R - regression model

When inputting the table containing the data points into desmos, I was able to use their regression tool to develop a regression model with the sine function as well. I decided to leave out the d variable as it pertains to the context of oscilloscope methodology, that I discussed earlier in Section 4, this also made more sense to perform a comparative analysis between the analytical and regression model, as there are the same number of variables to discuss. Therefore desmos would calculate for the vertical dilation (A), horizontal dilation (B) and horizontal shift (C).

$$y_1 \sim A \sin\left(B\left(x_1 + C\right)\right)$$

STATISTICS

$R^2 = 0.9618$

RESIDUALS

e_1

PARAMETERS

$A = 0.197973$ $\quad B = 11.0368$

$C = 0.0141732$

Figure 12. Parameter values as generated by the regression tool on desmos

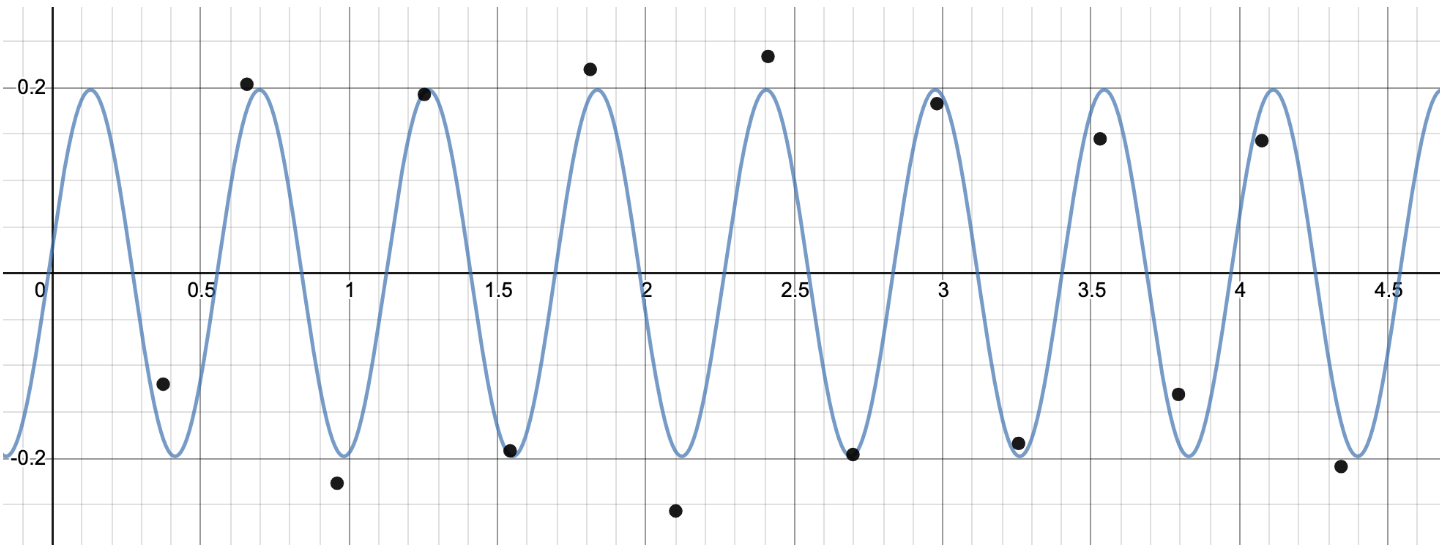


Figure 13. Graphical representation of R=0.197973sin(11.0368(x+0.0141732))

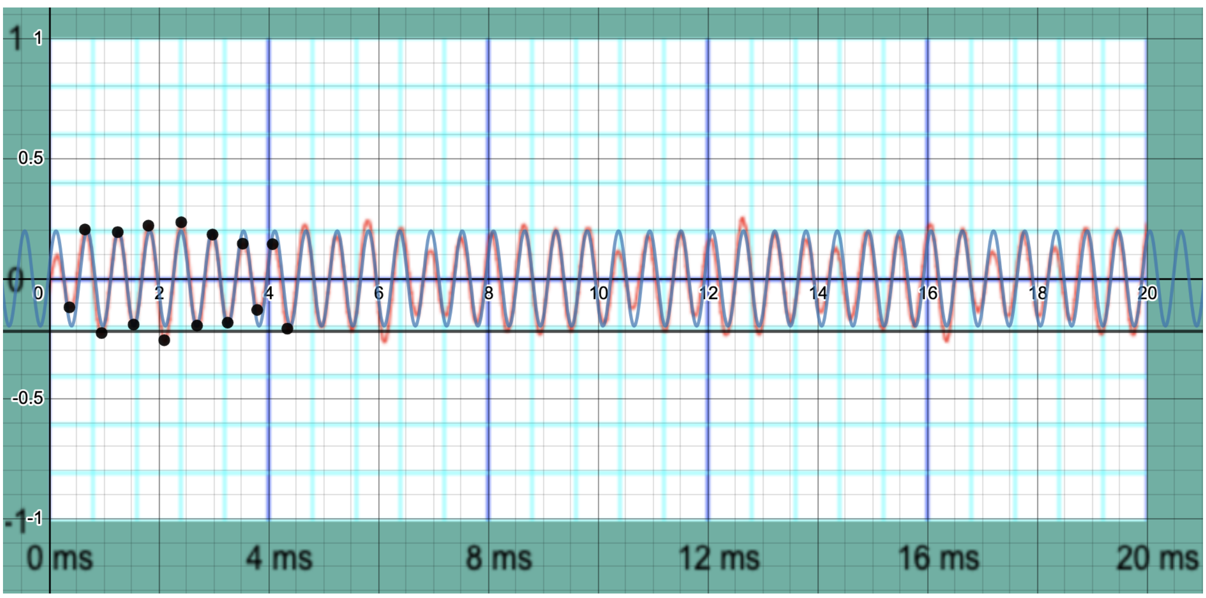


Figure 12 below shows results of the regression based parameters as calculated using desmos and figure 13 is the graphical representation of the function. This would result in Model R, which like Model M, has a domain of (0,20), as it only accurately represents values within the image and has a maximum of 0.198 and a minimum of -0.198.

$$\therefore R = 0.197973 sin(11.0368(x + 0.0141732))$$

Section 6. Evaluation

Comparing parameters

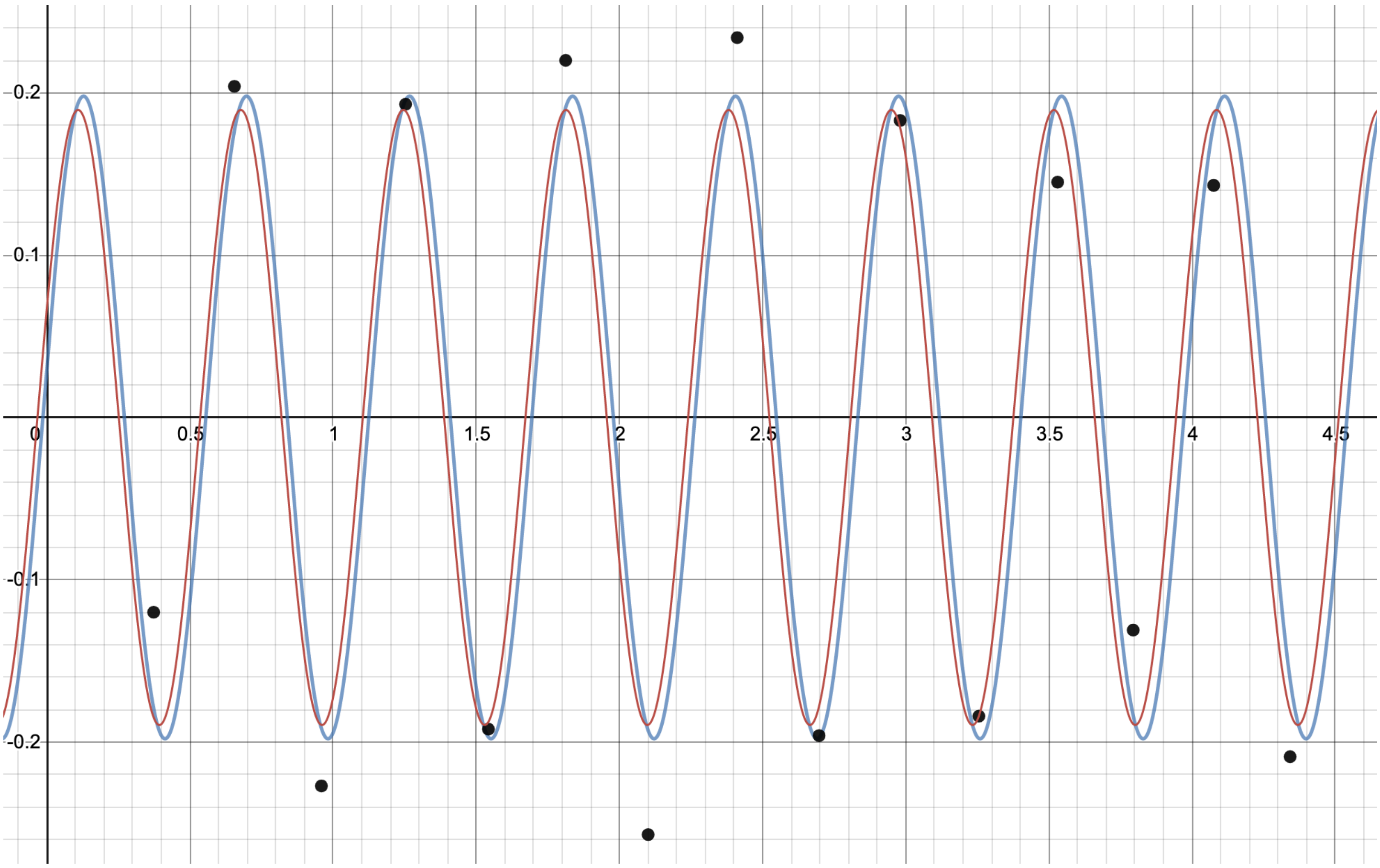


Figure 14. Graphs of Model M and Model R in the same plane, where red is model M and blue is model R

$M = 0.1895sin(\frac{\pi}{0.284}(x + 0.0328331421098$ $R = 0.197973sin(11.0368(x + 0.01417$

As illustrated in figure 14, Models M and R are quite similar, the most apparent differences lie in the vertical dilation and the horizontal shift.

When observing the parameters of the models itself, the **vertical dilation**, A, stands out as being very similar, with the difference at about only 4% of the model values as calculated below, this shows although the difference exists, it is very minimal. Additionally, the errors caused by vertical dilation stay constant over time as it alters the y values the same over time, therefore the errors do not increase.

Model M, A=0.1895
$\frac{0.197973-0.1895}{0.1895}$= 0.04471240106 = 4.47%

Model R, A=0.197973
$\frac{0.197973-0.1895}{0.197973}$= 0.04279876549 = 4.28%

Likewise, the **horizontal dilation**, B, between the two models are also quite close to each other, based on the calculation below, the B value of Model M is about 11.0619 when converted into decimals, whereas it's 11.0368 for Model R. This results in a difference of 0.0251, which is about 0.23% of the model value. However, unlike the vertical dilation, the difference between Model M and R will increase over time as the B value will alter the x values differently, therefore, the increasing errors will be a result of the difference regardless of how minimal it is.

Model M

$B = \frac{\pi}{0.284} = 11.0619459633 = 11.0619$

$\frac{0.0251}{11.0619459633} = 0.002269040193 = 0.23\%$

Model R

$B = 11.0368$

$\frac{0.0251}{11.0368} = 0.002274209916 = 0.23\%$

Upon this, I did realize that I could have made the analytical B value more generalized and took all the data points into consideration, which would involve doing the same solving for C for all the points (except only using the x_value), then finding the average C value which would be the horizontal shift in the equation.

Lastly, the **horizontal shift** essentially continues the trend. The difference between the two models relative to the period is 3.29%, which is a fairly small error. That is apparent in figure 10, which is the very slight horizontal shift difference between the two models. Like the vertical dilation, the errors due to this will also remain constant, only altering the x values the same way over time. As a result, this will not cause increasing errors.

$\frac{0.0328331421098-0.0141732}{0.568} = 0.03285201076 = 3.29\%$

Evaluating residuals

The analysis I conducted above did provide interesting insights between the Models themselves, but not as meaningfully between the Models and the data. In order to perform a more accurate and analytical comparative analysis on the relation of the two models to the data, I conducted a least-squares residual method to test for their accuracies. This regression method takes the sum, S_n, of the squares of the difference between predicted value and actual value. S_n is described with the following,

$$S_n = (h_1-\hat{h}_1)^2+(h_2-\hat{h}_2)^2+(h_3-\hat{h}_4)^2+...+(h_n-\hat{h}_n)^2$$

To perform the least-squares residual method without the manual hassle that is prone to errors and inconsistencies, I modified a piece of code I previously utilized for the same purpose, just a different root equation - quadratic. The code has been annotated and is as follows:

```
import math

a= #A value
b= #B value
c= #C value

x_list=[0.373,0.655,0.959,1.253,1.542,1.812,2.100,2.411,2.697,2.980,3.255,3.530,3.794,4.075,4.342]
y_list=[-0.120,0.204,-0.227,0.193,-0.192,0.220,-0.257,0.234,-0.196,0.183,-0.184,0.145,-0.131,0.143,-0.209]
results=[]  #a list to keep track of predicted y values
h_list=[] #a list to keep track of the squared differences between each actual and predicted y value

for i in range(15):
       y=a*math.sin(b*(x_list[i]+c)) #sine root equation excluding d variable (vertical shift)
```

```
        results.append(y)

for i in range(14): #taking the ith term of the y (actual) and result list, then subtract
        y_=y_list[i]
        r_=results[i]
        h=y_-r_
        h_list.append(h**2) #squaring the difference to remove any negative values

s=sum(h_list) #taking the sum of all the squares of differences
print('least squares result =', s)
```

Model M analysis	Model R analysis
$M = 0.1895sin(\frac{\pi}{0.284}(x + 0.0328331421098$	$R = 0.197973sin(11.0368(x + 0.0141732))$

```
import math

a= 0.1895
b= math.pi/0.284
c= 0.0328331421098

x_list=[0.373,0.655,0.959,1.253,1.542,1.812,2.100,2.411,2.697,2.980,3.255,3.530,3.794,4.075,4.342]
y_list=[-0.120,0.204,-0.227,0.193,-0.192,0.220,-0.257,0.234,-0.196,0.183,-0.184,0.145,-0.131,0.143,-0.209]
results=[]
h_list=[]

for i in range(15):
    y=a*math.sin(b*(x_list[i]+c))
    results.append(y)

for i in range(15):
    y_=y_list[i]
    r_=results[i]
    h=y_-r_
    h_list.append(h**2)

s=sum(h_list)
print('least squares result', s)
```

```
least squares result = 0.02290003359198007
```

```
import math

a= 0.197973
b= 11.0368
c= 0.0141732

x_list=[0.373,0.655,0.959,1.253,1.542,1.812,2.100,2.411,2.697,2.980,3.255,3.530,3.794,4.075,4.342]
y_list=[-0.120,0.204,-0.227,0.193,-0.192,0.220,-0.257,0.234,-0.196,0.183,-0.184,0.145,-0.131,0.143,-0.209]
results=[]
h_list=[]

for i in range(15):
    y=a*math.sin(b*(x_list[i]+c))
    results.append(y)

for i in range(15):
    y_=y_list[i]
    r_=results[i]
    h=y_-r_
    h_list.append(h**2)

s=sum(h_list)
print('least squares result, s)
```

```
least squares result = 0.02123837628648293
```

Through the code above, I was able to generate the following least-squares residuals:

$$R_M = 0.02290003359198007$$

$$R_R = 0.02123837628648293$$

Although in my previous analysis, I established that there was a high similarity between the analytical and regression model, it did still come as a surprise to me that the residuals were also quite close, with Model R only being 0.00166 less. Furthermore, when analyzing from a relativistic perspective, I found that the R_M is only larger by a factor of less than 1.01, which again is a great position to have an analytic model be relative to the regression model.

Afterall, I did make a lot of generalizations when creating the analytical model, which I expected to have caused slightly better results than not generalizing, but at the same time, I did not expect for it to achieve fairly close accuracy as the regression model.

In terms of the data, I did notice when looking back at the figure 14 that some of the points (at least 5 points) were hardly represented by either models, this should signify higher residuals, which I technically did not get. To be fair, the dataset already has low values (≤1) to begin with, which would automatically lower the results of the least squares calculation to ≤1 too, as anything ≤1 squared will remain ≤1.

$$\frac{0.02290003359198007}{0.02290003359198007} = 1.0782384342 = 1.01$$

This likely gave the illusion of lower residuals, which did give me the impression that this method may not be the best residual method to utilize. Although it is helpful for relative comparisons, it may not be suitable for absolute comparisons. Alas, according to the residuals, the regression model was still objectively the best fit and henceforth will be used for the following sections

Section 7. Real life predictions

When mathematical models are built to model and represent real world phenomena, it is important to know the extent at which these models are useful for. Therefore, In this section, I will be putting Model R to the test through several ways, to conclude its validity.

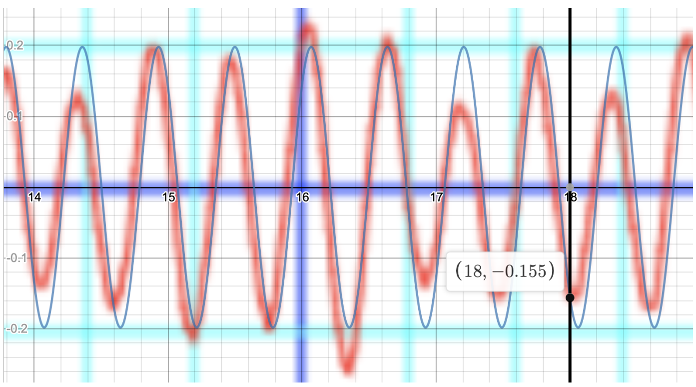
<table>
<tr>
<td>

The first situation I am testing is when **x=18ms**, would the predicted result of y be logical and at any extent comply with the actual data?
The following is the result of model R as calculated by desmos,

$$R = 0.197973sin(11.0368(18 + 0.0141732)$$

$$R =- 0.154891528335$$

This result is logical, although still inaccurate and can easily be verified by comparing it with the original image:



</td>
<td>

The second situation I am testing is looking for the times when the intensity of the sound is y=-0.22, for 15<x<18.

Based on Model R, these points would not exist as they are not even included in Model R's domain

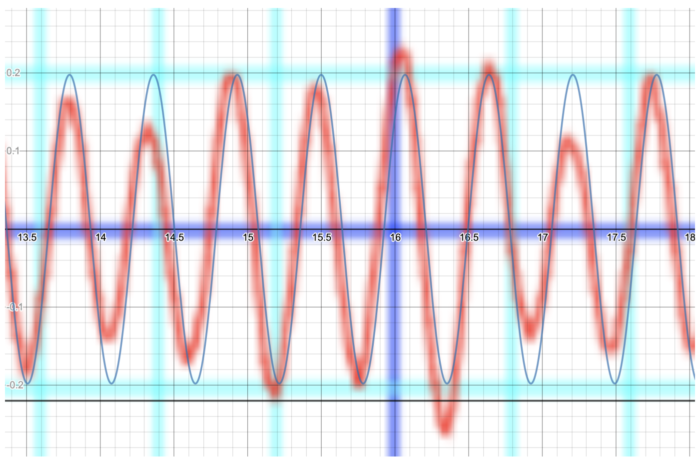


Figure 16. The black line is y=-0.22, while blue is Model R

</td>
</tr>
</table>

Figure 15. The black line is x=18, while blue is Model R At x=18 the local minimum does fall short of what the function predicts, therefore the actual y value would also be slightly closer to the minimum.	However, when observing the same point on the image from figure 1, it is clear that y=-0.22 is within the domain of the actual data, with two of points that would satisfy the question within the range.
Third, I am testing for negative values of x using x=-5ms as an example. Again by using desmos, I found that R would be equivalent to $R = 0.197973sin(11.0368(-5 + 0.0141732$ $R = 0.197728527172$ For this situation, it's difficult to determine whether it is accurate since the oscilloscope data does not extend to this x value, nor any other negative x value. This is likely for a good reason. In the context of an oscilloscope, it only detects sound over time, which can not be negative, unless this perspective is shifted. Therefore, although it could be accurate, testing for x=-5 or any other negative value is illogical.	The last piece of information to test is when x= a larger value that was not covered in the image, let alone the data set. I am using x=1000ms to test the following. $R = 0.197973sin(11.0368(1000 + 0.014173$ $R =- 0.102028558606$ Although this answer could be somewhat logical, it would certainly not be precise. This is largely due to important contextual components. The music that the data snapshot originated from had several wave patterns that highly vary. This means that a function that models a tiny snapshot of the music file certainly can not in any way accurately predict the other parts of it due to uncertainty. The only way this can plausibly occur is if the same pitch and intensity is being sustained for the entirety of time, for this model to actually be valid transcending this tiny snapshot.

Based on these experiments, it is suffice to say that the regression model can only sufficiently and logically represent and calculate for the points within its domain of 0<x<20. Even then, there are points are have been left out within the domain as they are out of range of the model to be detected

Section 8. Accuracy

Overall I approached all the steps within this report very carefully and attempted to maintain the highest degree of precision, keeping at least 3-4 significant figures and 3 decimal places for consistency purposes throughout the different parts of this report, both during the development of the models and well as in the evaluation. For the developmental parts, due to the need to transfer these values around different mediums and use them for different purposes, I couldn't necessarily conserve the detailed values effectively, nor was there a grave need to do so most of the time. Therefore I generally stuck with 3-4 significant figures and around 3 decimal points. As for the evaluation sections, especially with the regressions, a lot of details were able to be preserved. The python code especially, takes into account very precise values and does not require large complicated manual calculations, which helps reduce inaccuracy lost due to transformations under various mediums. However, at times I would end up with numbers that only contain only two significant figures, ie. in section 6 as I am calculating for the percentages.

Unfortunately, there were a few steps in the way that were inevitable and possibly contributed some downsides to the accuracy of this report or the integrity of its contents. Just to name and elaborate on a few:

- I was not able to gain any sort of I numerical data for the raw oscilloscope data as it was a limit of that technology. Therefore all the point selection and modifies had to all be done by hand, this makes the data slightly inaccurate since the beginning
- For defining the maximum and minimum values for Model M back in section 4, I chose to proceed by average all the minimum and maximums in those respective groups. Although this type of generalization could have created a better fit model in terms of having less residuals, the premise that some data points will not be included in the domain of a model whose purpose is to represent it, is quite strange and illogical, to say the least.

Section 9. A better fit

After all of these calculations and evaluations, it does open the possibility that there was another function which is likely much more accurate and encompassing for the overall data. Based on what I have personally studied or came across about this topic, it seemed like composite trig functions are likely the best. Since any and every wave can be modelled using the composite of several other waves, it implies that it can do the same for my oscilloscope data. This way, the model would not have to restrict itself into very few parameters, which is a great option for irregular sinusoidal data like those from oscilloscopes. When this is the case, I won't have to experience the issue of having many points deemed out of domain for the sake of generalization. Instead can actually be considered just as much as the rest.

Even then, the best these composite functions can perform is for the domains contextually restricted by the oscilloscopes, as once again, sound behaviour is unpredictable unless intended otherwise. As a general conclusion, these models are only good for modelling already existing data, but not necessarily to 'predict'.

Bibliography

“Desmos.” *Desmos*, Desmos, www.desmos.com/calculator/. Accessed 03 March 2021.

“Oscilloscope.” *Academo*, Academo, https://academo.org/articles/oscilloscope/. Accessed 03 March 2021.