

2018 年度

明治大学総合数理学部現象数理学科 卒業研究レポート

# 音の Fourier 解析

岸 朋美

2019 年 3 月 3 日

# 目次

第1章	はじめに p2
第2章	サンプリングについて p3-6
第3章	WAVE ファイルについて p7-8
第4章	音階について p9-10
第5章	Fourier 級数と離散 Fourier 変換 p11-12
第6章	解析する音と手順 p13-16
第7章	実験結果 p17-22
第8章	おわりに p23
	参考文献

# 第1章 はじめに

私が卒業論文のテーマを決めるにあたり、ギターやピアノを演奏することが趣味であることから、音と数学にどのような繋がりがあるのかを研究しようと考えた。そこで今回の卒業研究では、コンピュータで音を取り扱うために必要なサンプリングと呼ばれる技術や WAVE ファイルの内容、音階や離散フーリエ変換の知識をもとに **Mathematica** を利用して音の周波数解析を行った。

## 第2章 サンプリングについて

本来、音はすべての時刻で値をとる、「アナログ信号」の物理現象である。ただし、コンピュータはこうしたアナログ信号をそのまま取り扱うことができない。

コンピュータで音を取り扱うには、アナログ信号の音を離散的な時刻でしか値をとらない「デジタル信号」の音データに変換する必要がある。こうした処理を「AD 変換」と呼ぶ。



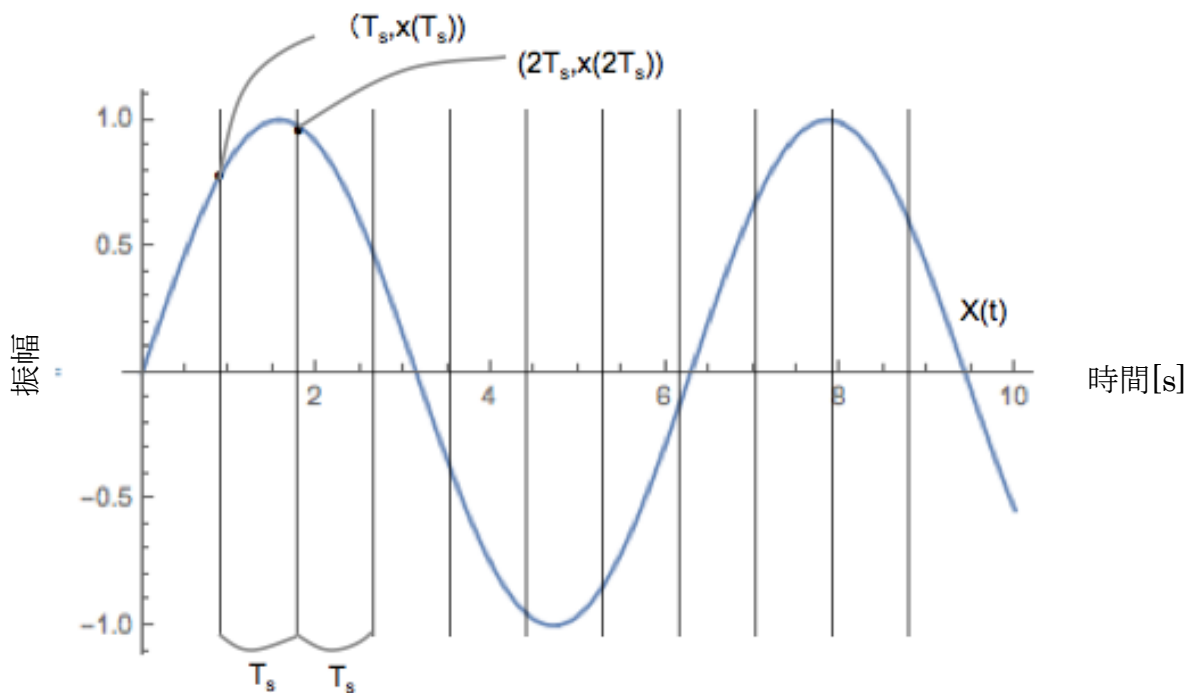
アナログ信号は、「標本化周期」と呼ばれる時間間隔でアナログ信号を読み取る「標本化」、標本化したアナログ信号を数として記録する「量子化」という二つの処理によってデジタル信号に変換される。

標本化は標本化周期を小さくすればするほど精度が高くなる。その性能は、標本化周波数によって比較される。標本化周期(サンプリング周期)とはサンプルをとる時間間隔のことであり、標本化周波数の逆数である。つまり

$$f_s = \frac{1}{T_s} \quad f_s: \text{標本化周波数} \quad T_s: \text{標本化周期}$$

という関係がある。

連続信号  $x(t)$  において正の数  $T_s$  を用いて  $x_n = x(nT_s)$  として  $\{x_n\}$  を求めることがサンプリングである。



標本化周波数を大きくすればするほど標本化の精度は高くなっていくが、あまりにも標本化周波数を大きくしてしまうと、デジタル信号に変換した時のデータ量が膨大になってしまう。そのため、標本化周波数をある程度に抑える必要がある。その指針になるのが「標本化定理」である。

標本化定理とは、関数  $x: \mathbb{R} \rightarrow \mathbb{C}$  の Fourier 変換

$$X(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} x(t) \exp(-i\omega t) dt$$

が

$$(\exists W > 0)(\forall \omega \in \mathbb{R}: |\omega| \geq W) X(\omega) = 0$$

を満たすならば、このような  $W$  を任意の一つ取って

$$T := \frac{\pi}{W}$$

とおくとき、

$$(\forall t \in \mathbb{R}) x(t) = \sum_{n=-\infty}^{\infty} x(nT) \frac{\sin \pi(n-t/T)}{\pi(n-t/T)} = \sum_{n=-\infty}^{\infty} x(nT) \text{sinc}[\pi(n-t/T)]$$

が成立する。つまり、 $\frac{W}{2\pi}$  以上の周波数成分を含まない信号は、サンプリング周波数  $f_s := \frac{1}{T_s} =$

$\frac{W}{\pi}$  でサンプリングした離散信号から復元できる。言い換えると  $f_{\max}$  以上の周波数成分を含まない信号は、 $2f_{\max}$  以上の標本化周波数でサンプリングしたデータから復元できる。

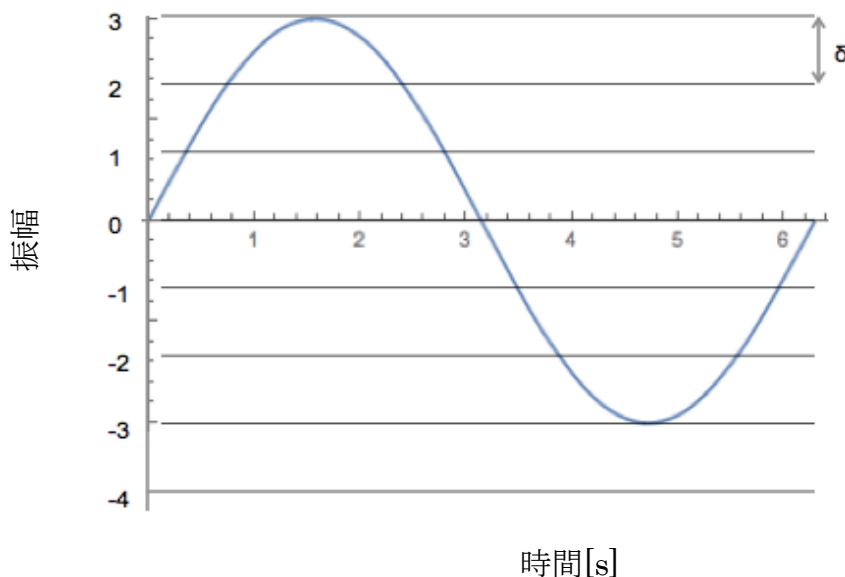
標本化周波数(単位:Hz)

192k	DVD-Audio
48k	デジタル放送のテレビ
44.1k	音楽CD
32k	FMラジオ、アナログ放送のテレビ
16k	AMラジオ
8k	電話

人間の聴覚は20Hzまでの音を聞き取る事ができる。標本化定理( $f_s \geq 2f_{max}$ )に照らし合わせると、標本化周波数を20kHzの2倍以上、すなわち40kHz以上に設定するのが妥当だと考えられる。

量子化とは、標本化が時間を離散化する処理になっているのに対して、振幅を離散化する処理になっている。振幅を区切るステップ数のものを「量子化精度」と呼ぶ。量子化精度は、データを記録するのに必要なデータ量にあたる。

$$N=2^Q \quad (Q: \text{量子化精度} \quad N: \text{ステップ数})$$



量子化精度を大きくすればするほどステップ幅  $\delta$  は小さくなり、波形の詳細な変化を落とさずにアナログ信号をデジタル信号に変換することができる。

### 量子化精度

24bit	DVD-Audio
16bit	音楽CD
8bit	電話

電話や AM ラジオではチャンネル数が一つの”モノラル”が普通である。ところで、人間には二つの耳があり、左右の耳に入る音の位相の差や音量の差などから、音がどこで鳴っているか音源の位置を把握している。このことから音楽を録音・再生する場合に利用したのが、チャンネルを2とするステレオである。

### 第3章 WAVE ファイルについて

音データをコンピュータに記録するための標準フォーマットとして一般的に利用されているのが、「WAVE ファイル」である。WAVE ファイルに録音された音データは様々な App から再生可能である。



fmt チャンクと data チャンクという二つのチャンクを格納するのが WAVE ファイルの基本構造になっている。fmt チャンクには標本化周波数や量子化制度など音データに関する情報が記述される。data チャンクには音データそのものが記録される。以下の表は WAVE ファイルのパラメータをまとめたものである。(青木[3])

パラメータ	サイズ(byte)	内容
riff_chunk_ID	4	'R' 'I' 'F' 'F'
riff_chunk_size	4	36+data_chunk_size
file_format_type	4	'W' 'A' 'V' 'E'
fmt_chunk_ID	4	'f' 'm' 't' ' '
fmt_chunk_size	4	16(リニアPCMならば)
Wave_format_type	2	リニアPCMは1
Channel	2	モノラルは1,ステレオは2
samples_per_sec	4	標本化周波数
bytes_per_sec	4	Block_size*sample_per_sec
block_size	2	bits_per_sample*channel/8
bits_per_sample	2	量子化精度(ビット数が8か16)
data_chunk_ID	4	'd' 'a' 't' 'a'
data_chunk_size	4	音データの長さ*channel
data	data_chunk_size	音データ

パラメータの意味を説明する。



- それぞれチャンクの先頭には、チャンクの ID とサイズが記述されており、いずれも 4byte の情報をになっている。riff\_chunk\_ID の内容は 'R' 'I' 'F' 'F' であり、4byte。また riff\_chunk\_size は RIFF チャンク ID と RIFF チャンクサイズを合わせた 8byte を除き、残りを数えた 36+data\_chunk\_size という内容の 4byte。
- file\_format\_type の情報は 'W' 'A' 'V' 'E' の四文字であるため、4byte。
- 同じように fmt\_chunk\_ID、fmt\_chunk\_size も各々 4byte ずつ。
- wave\_format\_type は音データの形式を表しており、PCM の場合は 1 である。サンプリングされたデジタル信号の音データを、専門用語で「PCM(Pulse Code Modulation)」と呼ぶ。
- channel はモノラルで 1、ステレオで 2 である。
- samples\_per\_sec は Hz を単位とする標本化周波数である。
- bytes\_per\_sec は一秒間の音データを記録するのに必要なデータ量を byte 単位で表したもので、内容は上記の通り。また、block\_size は一時刻の音データを記録するのに必要なデータ量を byte 単位で表したもの。1byte は 8bit に等しいため、block\_size は 8bit モノラルでは 1、8bit ステレオでは 2、16bit モノラルでは 2、16bit ステレオでは 4 になる。
- bits\_per\_sample は bit を単位とする量子化精度である。8bit の場合は 8、16bit の場合は 16 になる。

## 第4章 音階・音律について

音階とは楽曲のうちに用いられる音を整理して高さの順に配列したものであり、音階を構成する諸音の音高関係を数理的に規定したものを音律と呼ぶ。

西洋音楽の音階、ドレミファソラシドの音律はどのように決めるのか。人間に音がどのように聞こえるかを理解するには、波形だけでなく周波数特性を観察することが重要なポイントである。ここでは三つの音律を紹介する。オクターブとは西洋音楽における完全8度音程(7つ先の、同じ名前のついている音の段階)である。音の高さがオクターブ高くなると周波数はちょうど2倍になる。(1)から(3)ではオクターブの振動体の長さの比も比較している。

### (1)ピタゴラス音律

ピタゴラスは各々の音の弦の長さを整数比で表し、完全なものにしようとした。現在のドレミファソラシドに相当する音階があり、弦楽器に音階に応じた長さの弦を張り、得られたものがピタゴラス音律と呼ばれる。以下の比率は振動体の長さの比である。

- ◇ ド:レ=9:8 = 1.125:1
- ◇ ド:ミ=81:64 = 1.267:1
- ◇ ド:ファ=4:3 = 1.352:1
- ◇ ド:ソ=3:2 = 1.5:1
- ◇ ド:ラ=27:16 = 1.686:1
- ◇ ド:シ=243:128 = 1.898:1
- ◇ ド:上のド =2:1

### (2)純正律

16世紀の中頃、ツァルソーノが定めたものである。ピタゴラス音律は2と3の素数からできる整数比の音律だが、ツァルソーノは5という素数を含めて整数比で音律を作ることを提案した。

- ◇ ド:レ=9:8 = 1.125:1
- ◇ ド:ミ=5:4 = 1.25:1
- ◇ ド:ファ=4:3 = 1.333:1
- ◇ ド:ソ=3:2 = 1.5:1
- ◇ ド:ラ=5:3 = 1.667:1

$$\diamond \text{ド:シ} = 15:8 = 1.875:1$$

$$\diamond \text{ド:上のド} = 2:1$$

### (3) 12平均律

周波数次第で、sin 波はどのような高さの音にもなるが、音階を考慮して音の高さをコントロールすると、sin 波を使って音楽を演奏することができる。平均律は転調が容易であるという特徴を持つ。現代においては、特別な理由がない限り、平均律が用いられる。ドから上のドまでの一オクターブを 12 等分して、周波数が等比数列になるように音階を定められており、隣り合った周波数の比率は  $2^{1/12}$  倍になる。(左から A, A#, B, C, C#, D, D#, E, F, F#, G, G#, A であり、A はラ、B はシ、C はドの順である。)

$$\diamond \text{ド:レ} = 2^{2/12}:1 = 1.122:1$$

$$\diamond \text{ド:ミ} = 2^{4/12}:1 = 1.260:1$$

$$\diamond \text{ド:ファ} = 2^{5/12}:1 = 1.335:1$$

$$\diamond \text{ド:ソ} = 2^{7/12}:1 = 1.498:1$$

$$\diamond \text{ド:ラ} = 2^{9/12}:1 = 1.682:1$$

$$\diamond \text{ド:シ} = 2^{11/12}:1 = 1.888:1$$

$$\diamond \text{ド:上のド} = 2:1$$

## 第5章 フーリエ級数と離散フーリエ変換

与えられた関数を有限個または無限個の既知の関数の和として表現することは級数展開と呼ばれている。級数展開はその収束性に解析学に多くの興味深いテーマを提供するとともに関数の性質を調べたり、関数を近似したり、あるいは関数を具体的に計算する手段を与えるので極めて重要な手法である。

関数  $f(x)$  がある正の実数  $p$  に対して条件「どんな実数  $x$  に対しても  $f(x)=f(x+p)$ 」を満たすとき、関数  $f(x)$  は周期  $p$  の周期関数であるという。また、この条件を満たす実数  $p$  のうち、最小のものを、周期関数  $f(x)$  の基本周期という。

周期  $T$  の関数  $x(t)$  のフーリエ級数展開とは、三角関数の級数

$$\frac{a_0}{2} + \sum_{n=1}^{\infty} \left( a_n \cos \frac{2n\pi}{T} t + b_n \sin \frac{2n\pi}{T} t \right)$$

によって関数  $x(t)$  を表そうというものである。

実際、

$$a_n = \frac{2}{T} \int_0^T x(t) \cos \frac{2n\pi}{T} t \, dt$$

$$b_n = \frac{2}{T} \int_0^T x(t) \sin \frac{2n\pi}{T} t \, dt$$

とおくとき

$$x(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left( a_n \cos \frac{2n\pi}{T} t + b_n \sin \frac{2n\pi}{T} t \right)$$

が成り立つ。これを  $x$  のフーリエ級数展開とよぶ。

$a_n, b_n$  は  $x$  のフーリエ係数とよぶ。

また

$$c_n = \frac{1}{T} \int_0^T x(t) \exp\left(-\frac{2n\pi i}{T} t\right) dt$$

とおくとき

$$x(t) = \sum_{n=-\infty}^{\infty} c_n \exp\left(\frac{2n\pi i}{T} t\right)$$

が成り立つ。これを  $x$  の Fourier 級数展開とよび、 $c_n$  も  $x$  の Fourier 係数とよぶ。

$x$  を周期  $T$  の関数とするとき、 $N \in \mathbb{N}$  を固定して、

$$\omega = \omega_N := \exp\left(\frac{2\pi i}{N}\right)$$

さらに、 $j \in \mathbb{Z}$  に対して

$$t_j := j \frac{T}{N}, \quad x_j := x(t_j)$$

とおく。

$x_j$  は周期  $N$  の数列である。フーリエ係数  $c_n$  を台形公式で近似したものを離散フーリエ係数  $C_n$  とすると、

$$\begin{aligned} C_n &= \frac{1}{T} \sum_{j=0}^{N-1} x(t_j) \exp\left(-\frac{2\pi i n}{T} t_j\right) \frac{T}{N} \\ &= \frac{1}{N} \sum_{j=0}^{N-1} \omega^{-jn} x_j \end{aligned}$$

が成り立つ。 $C_n$  も  $n$  について周期  $N$  の数列である。

写像  $\{x_j\} \mapsto \{C_n\}$  を離散フーリエ変換という。

逆に

$$x_j = \sum_{n=0}^{N-1} C_n \omega^{jn}$$

も成り立つ。写像  $\{C_n\} \mapsto \{x_j\}$  を逆離散フーリエ変換とよぶ。

## 第 6 章 Mathematica により周波数を調べる方法

Mathematica には、あらゆる次元でのデータや関数、数列に対する全てのフーリエ変換や離散フーリエ変換が用意されている。ここでは Mathematica により周波数を調べた。

第 6 章では”do1.wav”のファイルを利用した。このファイルは、リニア PCM レコーダー (SONY PCM-A10) を利用して、静かな環境のもと、ギター の 5 弦 3 フレットを弾いた時の音を録音したものである。

以下では、録音したファイルを Mathematica を利用して周波数を調べる手順を述べている。

1. ファイルに読み書きするには作業フォルダを変更する必要がある。

```
SetDirectory["~/Desktop"]  
[ディレクトリの設定]
```

2. ”do1.wav”のファイルを読み込む。

```
snd = Import["do1.wav", "Sound"]  
[インポート]
```

3. 読み込んだ信号のリストを得る。

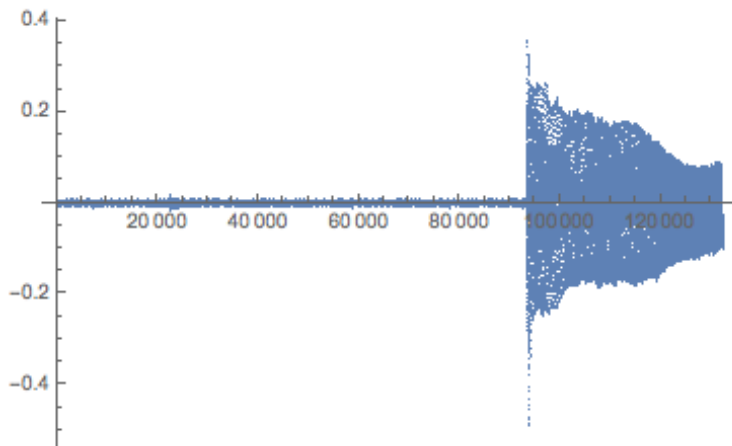
```
tbl = snd[[1, 1, 1]];  
Length[tbl]  
[長さ]
```

4. サンプリング周波数は 44100 となるはずである。

```
sr = snd[[1, 2]]  
44 100
```

5. sr から 3 秒分のデータを取り出してプロットしてみる。

```
tb = Take[tbl, {1, 3*sr}];  
[取り出す]  
g = ListPlot[tb, PlotRange -> All]  
[リストプロット] [プロット範囲] [すべて]
```

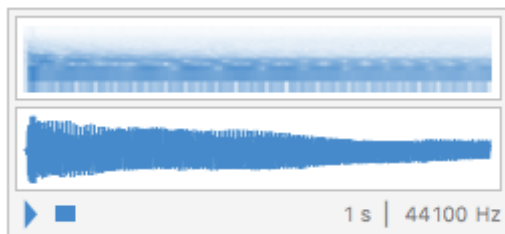


6.音が鳴り始める 93000 番目あたりの音から 1 秒分取り出して、3600 個分プロットした。

```
tb = Take[tbl, {93 000 + 1, 93 000 + sr}];
      [取り出す]
g = ListPlot[tb, Joined → True, PlotRange → {{1, 3600}, {-0.3, 0.3}}]
      [リストプロット] [点の結合] [真] [プロット範囲]
```

7.取り出したデータの音を鳴らす。

```
ListPlay[tb, SampleRate → sr]
[サウンドリス...] [サンプル数]
```

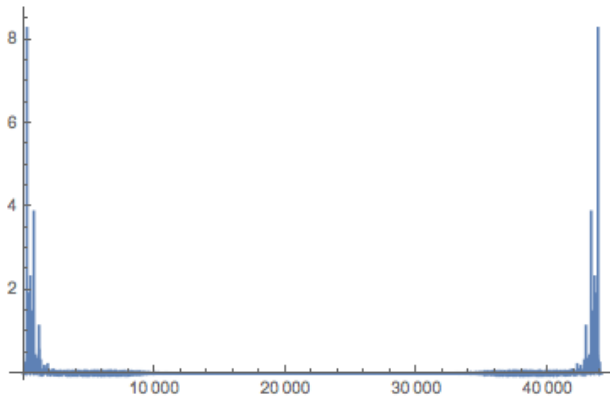


8.tb の離散フーリエ変換  $c$  を求め、絶対値をプロットする。これから周波数の分布がわかる。

$$x_0, \dots, x_{N-1} \Rightarrow c_0, \dots, c_{N-1}$$

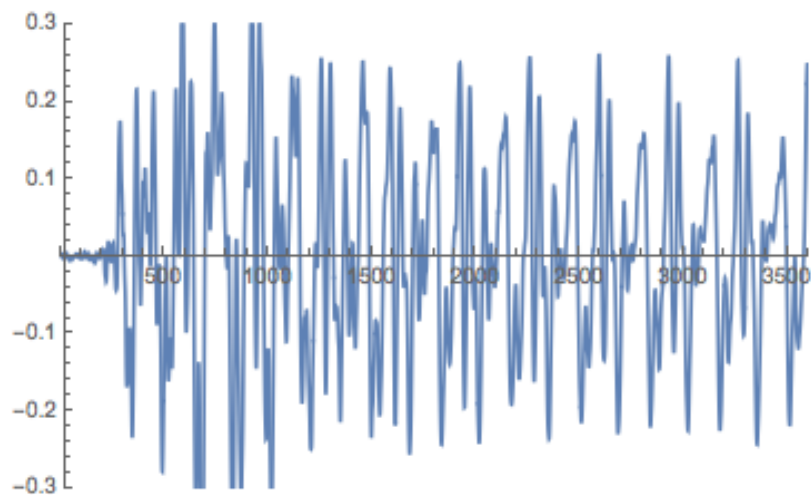
$c_0, \dots, c_{N-1}$  はそれぞれ  $x_0, \dots, x_{N-1}$  の離散 Fourier 係数である。Fourier[tb]は以上を計算する。

```
c = Fourier[tb];
      [フーリエ]
ListPlot[Abs[c], Joined → True, PlotRange → All]
[リスト...] [絶対値] [点の結合] [真] [プロット範囲] [すべて]
```



9. ギターの音が鳴り出してから 3600 個のサンプルの波形を見る。

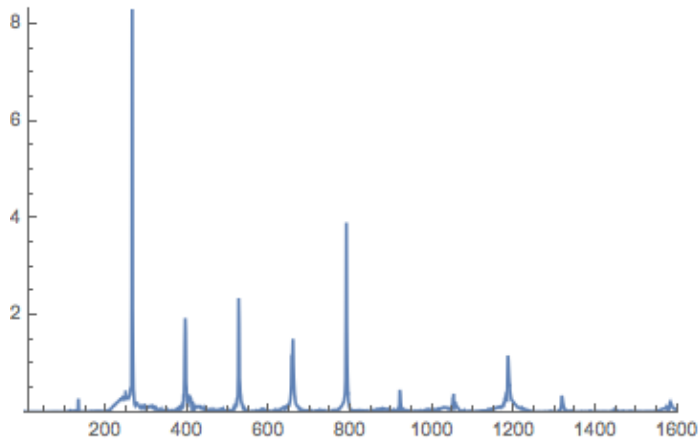
```
tb = Take[tbl, {93 000 + 1, 93 000 + sr}];
      [取り出す]
g = ListPlot[tb, Joined → True, PlotRange → {{1, 3600}, {-0.3, 0.3}}]
      [リストプロット] [点の結合] [真] [プロット範囲]
```



10. 音が鳴り出してから一秒間の信号を離散フーリエ変換して、1600 個の  $c$  を見る。

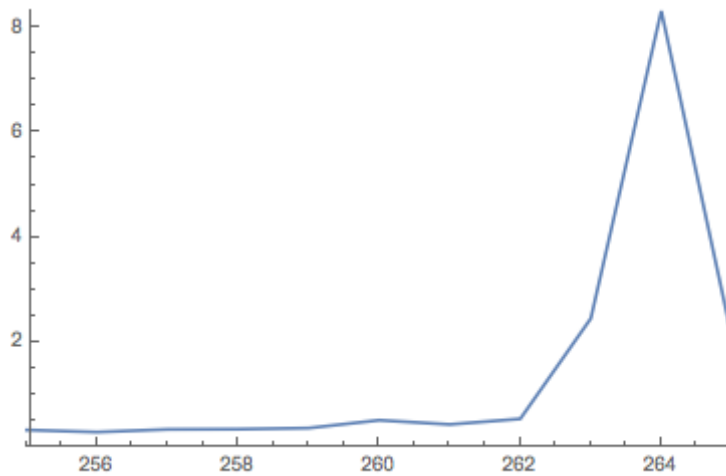
```
graph[c_, n1_, n2_] :=
  ListPlot[Abs[c], Joined → True,
    [リスト…] [絶対値] [点の結合] [真]
    PlotRange → {{n1, n2}, {0, Max[Abs[c]]}}]
    [プロット範囲] [最大] [絶対値]
graph[c, 1, 1600]
```





11. ピークが 255-265 で出ているので、さらに範囲を絞る。

`graph[c, 255, 265]`



- したがって、ギターのだの音、すなわち5限の3フレット目を押さえている時に弦を弾いた時、ピークは約 263Hz と言える。(Mathematia の Fourier 関数は最初のデータを 1 ではなく 0 から数えるので 1 ずれていることを考慮している。)

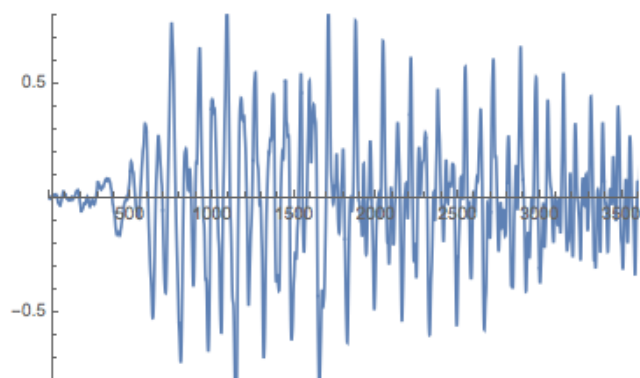
## 第6章 実験結果

- 今回採取した音は、ピアノの中央のド(鍵盤の左から数えて4つ目のド)から一オクターブ上のドまでの8音と、ギターのド(5弦の3フレットを押さえた時に弦を弾くと出る音)から一オクターブ上のドまでの8音と、ピアノの三音からなる協和音を採取した。

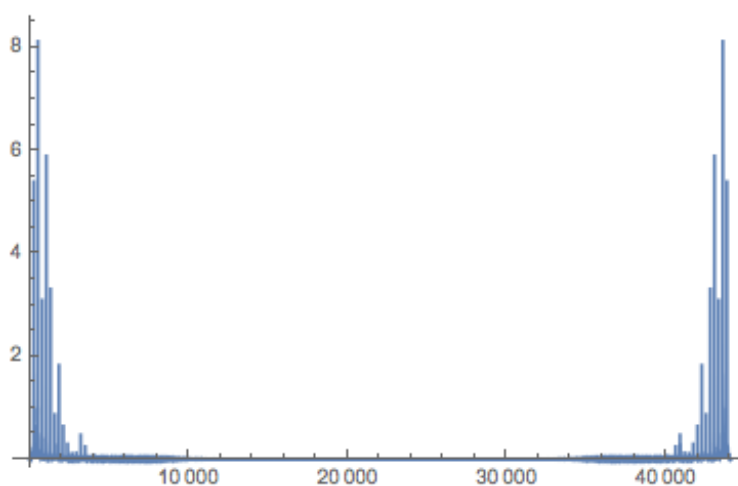
その中でも特にピアノのド～ミ、ソの音と協和音の実験結果について記載する。

### (1) ピアノのドの音

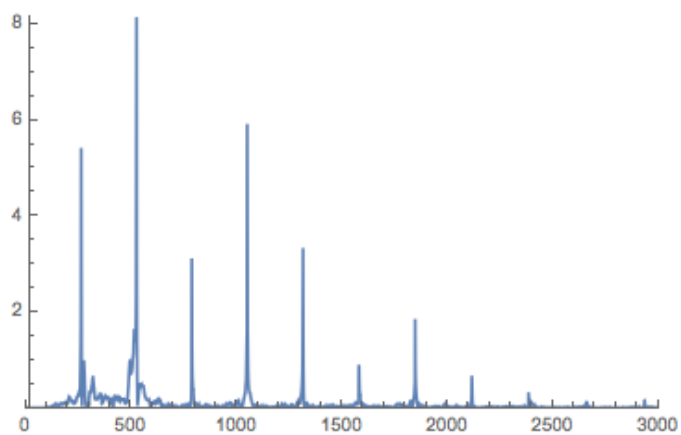
#### (a) 時間波形



#### (b) DFTした図



#### (c) スペクトル

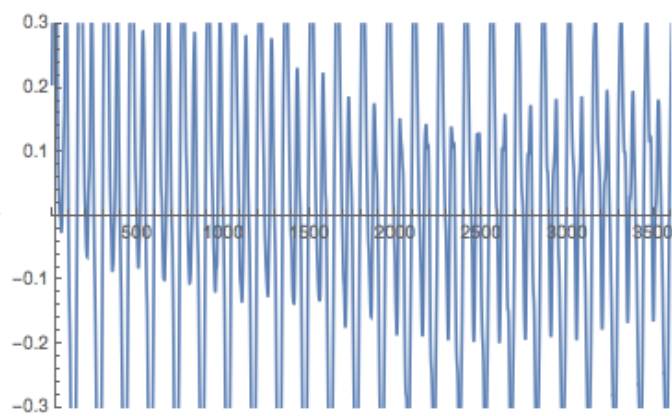


(d) ピークの位置

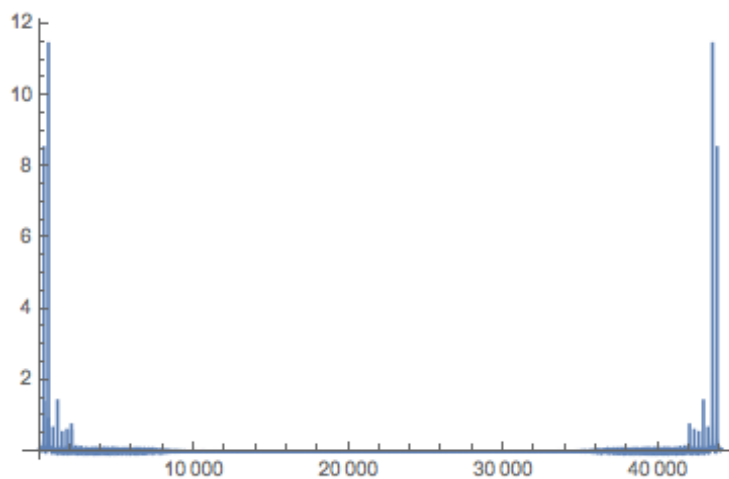
{262,525,788,1050}

(2) ピアノのレの音

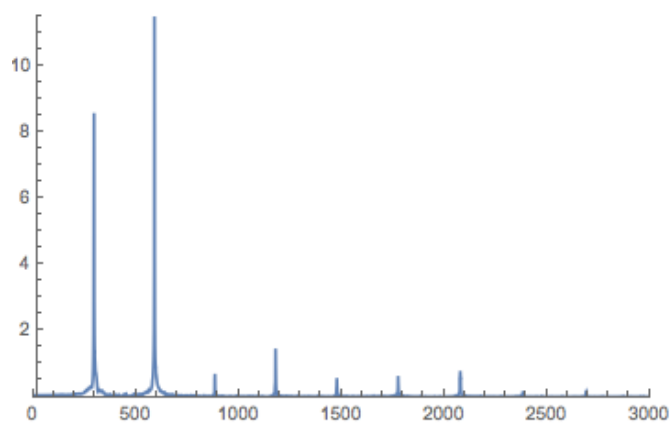
(a) 時間波形



(b) DFT した図



(c) スペクトル

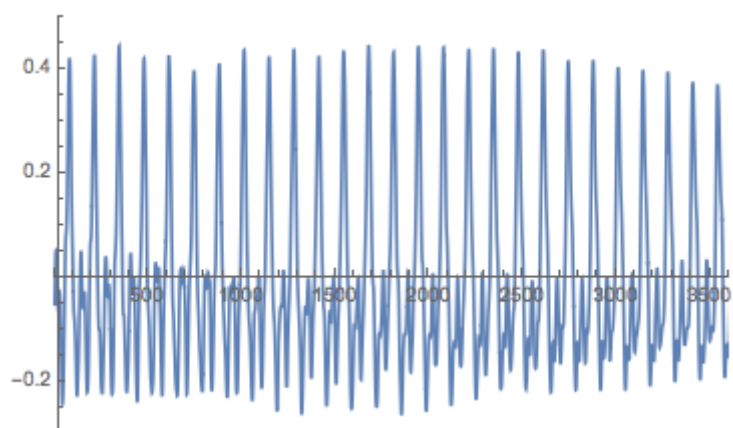


(d) ピークの位置

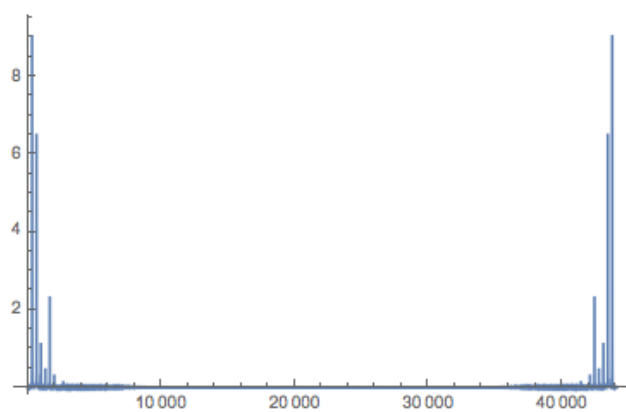
{294,589,883,1178}

(3) ピアノのミの音

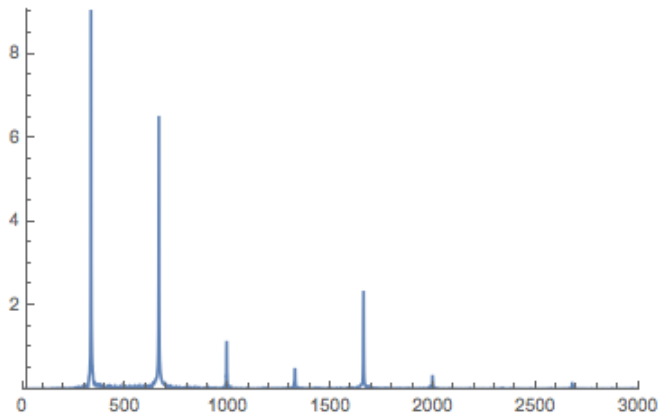
(a) 時間波形



(b) DFTした図



(c) スペクトル

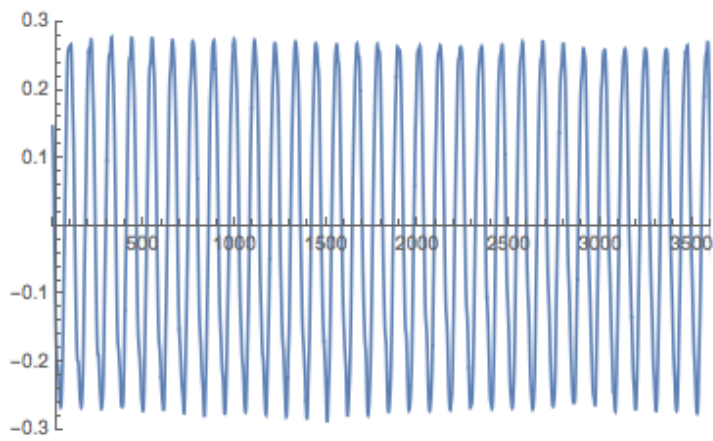


(d) ピークの位置

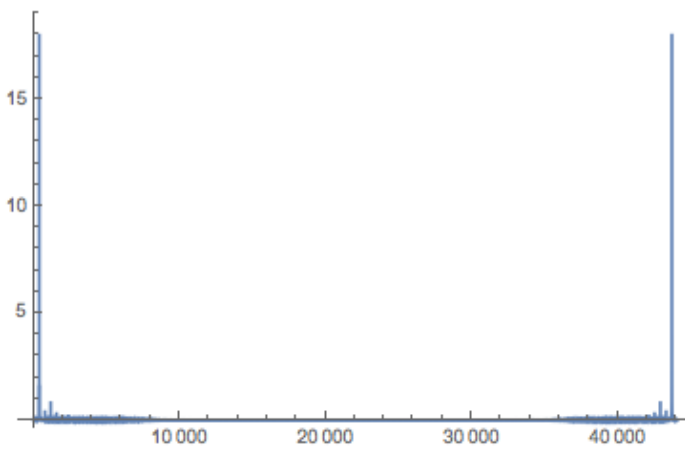
{331,661,992,1324}

(4) ピアノのソの音

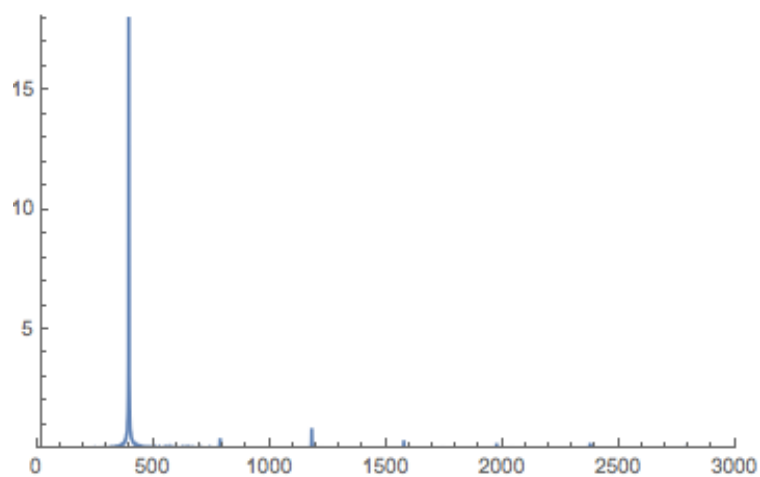
(a) 時間波形



(b) DFT した図



(c) スペクトル

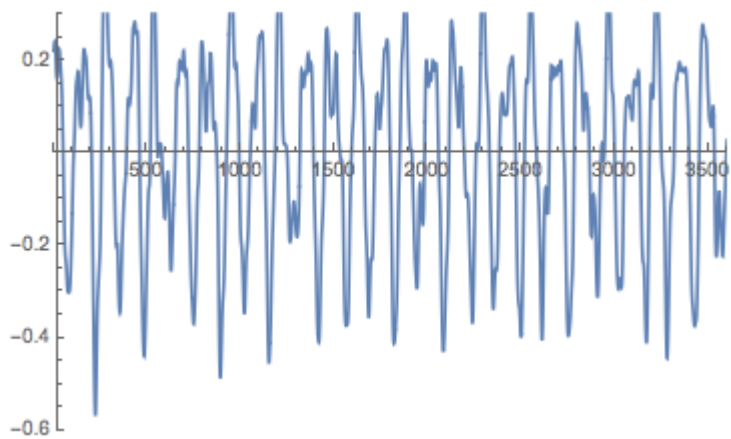


(d) ピークの位置

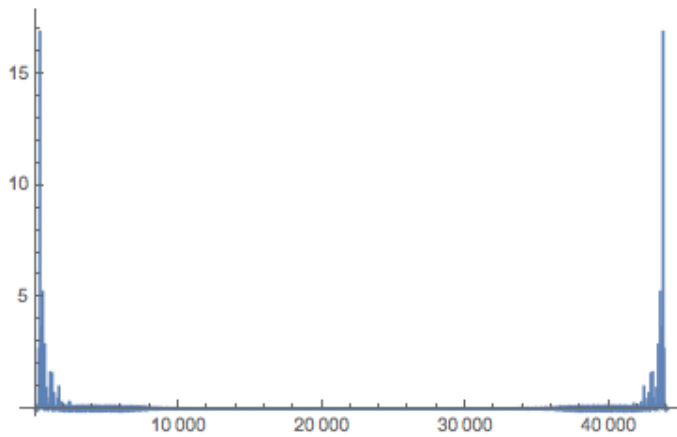
{393,785,1574,1974}

(5) ピアノのドミソを同時に鳴らした音

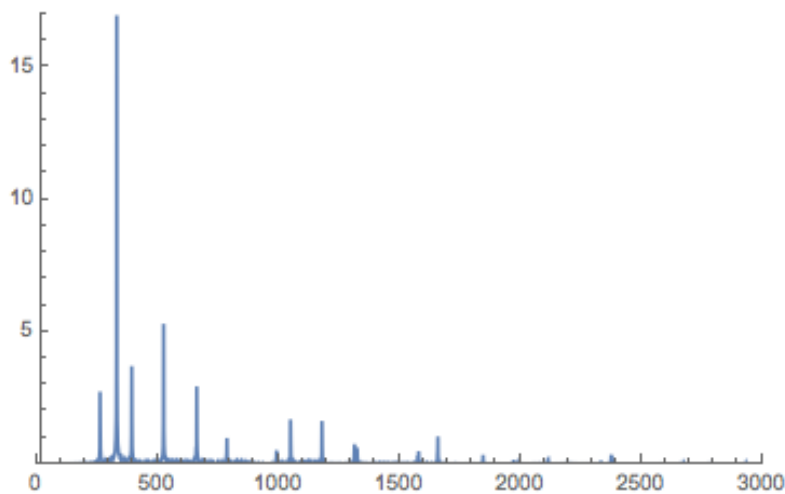
(a) 時間波形



(b) DFT した図



(c) スペクトル



(d) ピークの位置

{262,331,392,524,662,786}

- また協和音である、ピアノのドミソの周波数(262:331:392)に注目すると、単音で得たドとミとソの周波数(262:331:393)と近い値となった。また、ドの音の3倍音(788Hz)がソの2倍音(785Hz)とよく合っていることから協和していることがわかる。
- ド、レ、ミ、ソの単音の周波数を観察すると、どの音も倍音を持っていることがわかった。例えば、262Hzの2倍音と3倍音を計算すると、524Hzと786Hzであるが、実験結果では2倍音に近い525Hz、3倍音に近い788Hzを得ることができた。

## 第7章 おわりに

音と数学には深い繋がりがあることがわかった。さらにデータの種類を増やし、楽器ごとに音律と周波数がどのようなつながりがあるか調べることができたら面白かったと思う。

また、コンピュータで音を取り扱うために必要となる知識を学べたことと、それを実際に実行できたことが、とてもおもしろかった。音階は上記で紹介した三種類以外にもたくさん存在し、全てが数字で表せたことから、音律と数学は密接しているのだと実感した。



## 参考文献

- [1]「Fourier-Laplace 解析」 著 木村英紀 岩波書店 (1993 年 9 月 8 日)
- [2]「平均律について」 <http://stby.jp/heikinritu.html>
- [3]「サウンドプログラミング入門」 著 青木直史 技術評論社 (2017 年 9 月 16 日)
- [4]「数学と音楽～遠そうで近い関係～」 著 中村克己 明治大学数学科フレッシュマンズデー配布プリント (2012 年 4 月 21 日)
- [5]「音の Fourier 解析」 2013 年度 明治大学理工学部数学科 卒業研究レポート  
松山周五郎
- [6]「ギターの名の Fourier 解析」 2010 年 明治大学理工学部数学科 卒業研究レポート  
山田祐二