

第4章 音声信号の周波数を調べる実験

(ここは、実際に Mathematica を使って試すことを想定して説明してある。)

4.1 まずやってみよう

4.1.1 準備

- (1) この授業の WWW サイト¹ から、ギターのだの音の WAVE ファイル guitar-5-3.wav を入手し²、どこか適当なところ (デスクトップとかホームディレクトリとか) に保存する³。
- (2) Mathematica を起動して、カレント・ディレクトリを (1) のディレクトリにする (ホームディレクトリに保存した場合は必要がない)。
具体的なコマンドは、例えばデスクトップに保存した場合は

```
SetDirectory["~/Desktop"]
```

FileNames[] というコマンドで、カレント・ディレクトリにあるファイル名一覧が表示される。

4.1.2 guitar-5-3.wav の音を離散 Fourier 変換する

以下の実験の内容は、おおむね卒研のレポート (松山 [9]) に基づく。

```
snd=Import["guitar-5-3.wav"]
```

これでファイル guitar-5-3.wav を変数 snd に読み込む (snd は sound のつもり)。ボタンを押すと音が再生できる。

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

左チャンネルの音の数値データを tbl に代入した (tbl は table (表, 数表) のつもり)。

右チャンネルの音が欲しければ `tbl=snd[[1,1,2]]` とするか、あるいは `{tbl,rtbl}=snd[[1,1]]` として同時に代入する。ちなみに `snd[[1,2]]` はサンプリング周波数である。

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

44100 となるはずである (sr は Sample Rate のつもり)。音楽用 CD と同じ、44.1 kHz というサンプリング・レートで録音したことを示している。

¹<http://nalab.mind.meiji.ac.jp/~mk/fourier/>

²実は保存しなくても、`snd = Import["http://nalab.mind.meiji.ac.jp/ mk/fourier/guitar-5-3.wav"]` のように URL を指定して Import 出来るが、実行時にネット接続が必要なので、最初に保存しておくのが便利である。

³例えば、Safari では、`control` を押しながらクリックして、「リンク先のファイルを別名でダウンロード」を選択して、デスクトップを選択すれば良い。

```
tb = Take[tbl, {1, 3*sr}];  
g = ListPlot[tb, PlotRange -> All]
```

sr から 3 秒分のデータを取り出してプロットしてみた。

(サンプリング周波数が sr = 44.1 kHz なので、1 から 3*sr で、3 秒分のデータということになる。Take[] はリストから、指定した範囲のデータを取り出す関数である。)

```
tb = Take[tbl, {62800+1, 62800+sr}];  
g = ListPlot[tb, Joined -> True, PlotRange -> {{1, 1600}, {-0.3, 0.3}}]
```

音が鳴り始めるのは 62800 番目辺りからなので、そこから 1 秒分 (sr × 1 s = 44100 個のデータ) 取り出して、1600 個分 (1600/44100 ≒ 0.036 秒分) プロットしてみた。ここは色々試してみると良い。

```
ListPlay[tb, SampleRate->sr]
```

とすると、取り出したデータ tb の音を鳴らすことが出来る。

```
c = Fourier[tb];  
  
ListPlot[Abs[c], Joined->True, PlotRange->All]
```

tb の離散 Fourier 変換 c を求め、絶対値をプロットした。これから周波数の分布が読み取れる (はず)。Abs[] の代わりに Re[], Im[] としてみたり (分かりにくいけれど、 $C_n = \overline{C_{N-n}}$ の関係が見える?)。

```
(* n1~n2 の範囲で |c[[n]]| をプロットする。 *)  
graph[c_, n1_, n2_] := ListPlot[Abs[c], Joined -> True,  
    PlotRange -> {{n1, n2}, {0, Max[Abs[c]]}}]  
  
graph[c, 1, 1600]  
graph[c, 120, 140]
```

範囲を区切って表示することで、ピークを探してみた (左右対称なので左側だけで探す)。素朴に目で見えて探したが、プログラムを書いて自動化することも難しくないであろう。

ピークは 130 番目である。つまり $|C_{129}|$ が最大ということである。(リスト c の 1 番目の要素 c[[1]] は C_0 であり、リストの要素の番号と Fourier 係数のインデックスが 1 ずれていることに注意する。)

これはこのギターの音の基本周波数が 129 Hz (ドの周波数 131 Hz に近い — ぴったりでないのは、録音をした人 (研究室の学生) のチューニングが完璧ではなかったから) であることを意味する。

手近の楽器の音 (ピアノ, リコーダー, ...) を自分で録音出来る人は、試してみよう。

逆 Fourier 変換して元に戻るか? —

```
tb2=InverseFourier[c];  
Norm[tb-tb2]
```

実数データにするため、tb2=Re[InverseFourier[c]]; とする方が良いかもしれない。

余裕があれば、<http://nalab.mind.meiji.ac.jp/~mk/fourier/mathematica-memo/node9.html> も見てみよう (高い周波数の音をカットする実験)。