

## 目次

- パターン情報の表現 (1) — 標本化と量子化
- パターン情報の表現 (2) — データ圧縮と情報量

## ★2 パターン情報の表現 (1) — 標本化と量子化

パターンとはどんなものであるか、コンピュータ上のデータとして表現するにはどうするのかを考える。例として、音響信号とデジタル画像をとりあげる。以下では信号という用語が出てくるが、今回の授業の範囲では、信号とパターンは同じものと考えて差し支えない。いずれも、時間や空間にともなって変動する、何らかの情報を伝える量のことである。

信号: signal

### 2.1 音響信号の例

音は、空気などの振動によって伝わる。音響機器は、マイクロフォン (☆1) によって音を電気信号に変換したものを扱う。したがって、音響信号すなわち音のパターンは、「時間にもともなって変動する電圧」として表現される。「時間が**変数**で電圧が**値**である」ということもできる。例として実際の音響信号をグラフに描いてみると… (授業中に示します)。

☆1) マイク, microphone. 内部の振動板でとらえた空気の振動を電圧の変動に変換する。逆の変換をするのがスピーカ。

### 2.2 標本化と量子化

このように、音響信号は本来、変数 (時間) も値 (電圧) も連続量で表されるアナログ信号である。しかし、コンピュータ上のデータとしては、変数も値も離散的なデジタル信号として扱われる。アナログ信号からデジタル信号への変換 (AD 変換 (☆2)) は、**標本化と量子化**という二つの処理から成る。

アナログ信号: analog signal

デジタル信号: digital signal

■**標本化** 連続な「変数」を離散化する。「○○を1秒間隔で観測する」, 「××を1mおきに測定する」といったことをして変数をとびとびにする。その間隔のことを**標本化間隔** (または**標本化周期**) といい、その逆数を**標本化周波数**という。

☆2) 逆は DA 変換。

標本化: sampling

■**量子化** 連続な「値」を離散化する。例えば、1時間おきに測定した気温の値を四捨五入して整数値にする、あるいは5度刻みで段階分けする、といった操作のことである。値を段階分けする数を**量子化段階数** (または**量子化レベル**) という。量子段階数に2のべき乗を選ぶと、コンピュータでの扱いが便利である。

量子化: quantization

信号の分類としては、アナログ信号/デジタル信号の他に、変数が連続で値が離散なもの (☆3) や逆に変数が離散で値が連続なものもある。また、後述の画像の例からわかるように、変数や値が多次元の量であるものもある。

☆3) 例えば、変数: 講義開始からの経過時間, 値: 寝てる人の数。

**Q1.** 標本化周波数 100Hz のデジタル信号を考える。この信号 2 秒分はいくつの値から成ると考えられるか。

**Q2.** 量子化段階数が  $256 = 2^8$  であった場合、ひとつの値を表すには何 bit 必要か。

**Q3.** 時々刻々変化する気温を標本化間隔 30 分で標本化し、量子化段階数 64 で量子化して得られるデジタル信号を考える。この信号 24 時間分は、いくつの値から成るか。また、24 時間分のデータのデータ量は何 bit になるか。

## 2.3 デジタル画像の例

デジタルカメラで何かを撮影すると、平面上に一定の間隔で画素が並んだ画像が得られる。個々の画素は、明暗のみのグレイスケール画像なら 1 つ、カラー画像なら色を表現する 3 つの値 (☆4) をもっており、これを画素値という。画素値も離散的な量で表されるので、このような画像は、変数も値も離散なデジタル信号 (デジタル画像) である。変数は平面上の位置を表すから 2 次元であり、値はグレイスケールなら 1 次元、カラー画像なら 3 次元である。

実際のデジタル画像をみてみよう… (授業中に示します)。

**Q4.** あるデジタル画像の形式では、1 つの画素の色を赤緑青の 3 つの値で表しており、それぞれの値に 3bit を割り当てているという。この場合、何色を表現できるか。

画素: pixel

グレイスケール: grayscale or greyscale. モノクローム (monochrome) とも。

☆4) R (赤), G (緑), B (青) の 3 つの成分で表す方法がよく知られているが、他に色相・彩度・明度の 3 つで表すものなど色の表現法には様々なものがある。

## ★3 パターン情報の表現 (2) — データ圧縮

### 3.1 パターンのデータ量を見積もってみよう

パターンを表現するデジタルデータのデータ量がどれくらいになるのか見積もってみよう。CD に記録された音楽データの例、画像の例 (授業中に示します)。

### 3.2 データ圧縮とは

上で説明した例からもわかるように、パターンを表現するデジタルデータのデータ量は非常に大きくなりがちである。そのようなデータをそのまま記録したり通信したりするのは記憶容量や通信帯域が勿体ないので、データ圧縮を行うことが一般的である。

データ圧縮とは、あるデータが与えられたときに、それが含む情報の性質を損なわないでよりデータ量の小さいデータに変換する処理のことである (☆5)。あるデータをより小さいデータ量のデータに変換することを「圧縮する」といい、圧縮されたデータを元のデータに戻すことを「伸長する (あるいは展開する)」という。「符号化する/復号する」ということもある (☆6)。元のデータと圧縮したデータのデータ量を比べて、前者より後者が十分小さければ「圧縮率が高い」といい、そうでなければ (あまり変わらなければ) 「圧縮率が低い」という。

勿体ない: mottainai.

データ圧縮: data compression

☆5) データ圧縮の対象はパターンに限らないことに注意。多様な種類の情報に適用される、非常に重要な情報処理の方法である。

☆6) 本来、「符号化」には標本化・量子化のような処理も含む広い意味がある (第3回「よだんだよん」も参照) が、「圧縮」の意味で使われることも多い。ちなみに、「復号」は動詞なので「化」はつけない。俗に「解凍する」ということも。

### 3.3 データ圧縮手法の分類

データ圧縮の手法には様々なものがあり、いろいろな観点から分類することができる。例えば、全てのデータ圧縮手法は次の二つに分けられる。

**可逆圧縮** 逆が可能、つまり、圧縮したデータを伸長したら元のデータと完全に同じものが得られるようなデータ圧縮の方法。プログラムやテキストデータを対象とするなら一般にこちらを用いる。

**不可逆圧縮** 伸長しても元のデータを完全に再現できるとは限らない方法。その分、可逆圧縮よりも高い圧縮率を実現できる。音声や画像といったパターン等、情報の欠落や変化が多少あっても許容できる性質のデータに用いられる。

身近なところでは、コンピュータ上のファイルを圧縮するツール(☆7)は可逆圧縮を行なっている。静止画像のファイル形式では、PNGは可逆圧縮であるが、JPEGは不可逆圧縮である(☆8)。映像データの形式であるMPEGや、オーディオデータの形式であるMP3なども不可逆なデータ圧縮方式を採用している(☆9)。

可逆圧縮の手法は、次の二種類に分けることができる。

- **データ値の並び方を利用する手法:** 同じ値が連続している、近い値が連続している、文章中の単語のように一定の並びが繰り返し出現する、などといった性質を利用する。後述のランレングス圧縮や、LZ符号化(☆10)などが代表的。
- **データ値の現れ方の偏りを利用する手法:** 次回登場予定のエントロピー符号化(ハフマン符号化や算術符号化が代表的)など。

いずれの手法も、データの種類(画像であるとか音声であるとかテキストであるとか)によらず幅広く用いられる。不可逆圧縮したデータをさらに圧縮するのに用いることも多い。

一方、不可逆圧縮の手法は、対象とするデータの種類を(静止画像向け、音響信号向け、などのように)限定し、データの性質を利用して高い圧縮率を実現しているものが多い。可逆圧縮手法のように一般的な性質でもってそれらの手法を分類することもできるが、割愛する。

**Q5.** 自分の持っている画像データがどんな形式で保存されているか調べてみよう。ウェブ上の画像がどんな形式か調べてみよう。単純には、そのファイルの拡張子を見ればよい(☆11)。おそらく、JPEGやPNGが多いだろう。それぞれどのような画像に使われることが多いか、どのような画像に向いているかを調べてみよう。

可逆圧縮: lossless compression, 不可逆圧縮: lossy compression

☆7) `compress, gzip, bzip2` といったUNIX系OSのコマンドや、MS-DOSやWindowsで良く用いられるLHAなどのソフトウェア、ZIP形式を扱うツールなど。

☆8) PNG: Portable Network Graphics. JPEG: Joint Photographic Experts Group (この方式を作った組織の名前でもある)。JPEGは可逆手法と不可逆手法を組み合わせているので全体として不可逆。ただし、可逆圧縮のみ行うことも可能。

☆9) MPEG: Moving Picture Experts Group. MPEG-1, MPEG-2など様々な規格がある。DVD-VideoはMPEG-2をベースとしている。MP3はもともとMPEG-1のオーディオデータの規格。

☆10) 1970年代後半にZivとLempelが提案した手法。様々に改良され、現在も幅広く使われている。有名なLHAもこれを用いる圧縮ツールの一つ。

☆11) 拡張子とファイル形式との対応は、ネットや書籍で調べてみよう。ただし、拡張子が間違っている/わざと変えてある(初歩的なコンピュータウィルスなどの手口)こともあるが。

**Q6.** 計算機室の Linux 環境で、適当なファイルを圧縮してみよう。

```
$ ls -l hoge      ← 圧縮前のサイズ (バイト単位) を調べる
$ gzip hoge      ← hoge を圧縮. hoge.gz というファイルができる
$ ls -l hoge.gz  ← 圧縮後のサイズを調べる
```

伸長の仕方は、

```
$ man gzip
```

して調べよう。ファイルの種類（プログラムのソース、実行形式、JPEG 画像、etc.）によって圧縮率が違ったりするだろうか。

**Q7.** 画像を扱える適当なソフトウェア（フリーで手に入るものも多い）を用いて、適当な画像を様々な形式で保存してみよう。圧縮率をいじれるならば、ファイルサイズと見た目がどのように変化するかいろいろ試してみよう。

計算機室の Linux 環境なら、ImageMagick (画像処理ツール群) が使える。ImageMagick の `display` コマンドを使って

```
$ display hoge.jpg
```

とすれば画像 `hoge.jpg` を画面に表示できるし、そこからマウスクリックしてメニューを開いて画像形式を選択して保存することもできる。ImageMagick には他にも画像の大きさや形式を変換する `convert` などがある。