

14 相互相関

14.1 ベクトルの類似度

ベクトルの類似度を計算する方法はいくつかある。ベクトルを2次元の座標だとすると、2次元座標上の2つのベクトルは、2つの点を表わす。

ここで $w = (1, 1)$, $x = (3, 3)$, $y = (-1, 1)$ であるとする。

2つの点の距離は、2つの点の近さを表わす。

練習 47 2次元座標上に点 w, x, y を図示せよ。さらに、 w, x の2点間の距離(ユークリッド距離)と w, y の2点間の距離を求めよ。距離で見ると x, y のどちらが w と近いのか?

$w = (1, 1)$, $x = (3, 3)$, $y = (-1, 1)$ を位置ベクトルだと思える。位置ベクトルには向きがある。向きがどのくらい似ているか、を考えることもできる。

ベクトル間の角度は内積を用いて計算できる。 a と b の内積は $a \cdot b$ とあらわすことにする。

練習 48 2次元座標上に位置ベクトル w, x, y を図示せよ。さらに $w \cdot x$ と $w \cdot y$ を求めよ。内積で見ると x, y のどちらが w と似ているのか?

系列を考えるために、もう少し点数の多いベクトルを考える。

```
>> a=[1 5 -1 3];
>> b=[4 20 -4 12];
>> c=[-3 1 5 1];
>> plot(a); hold on; plot(b, 'r'); plot(c, 'g')
```

プロットを見ればわかるように、 b は、 a を4倍したものである。波として見れば、振幅が異なるだけで形は同じである。一方、 c は a, b とは違う形である。

練習 49 上記の a, b, c に関して、MATLAB で内積とユークリッド距離を求めよ。(内積を求める関数は `dot` である。また、ユークリッド距離を求める関数は `norm` である。) 求めた結果、`dot` と `norm` のどちらが波の形の類似性を求めるために適切か考えてみよ。

練習 50 適当な(同じ長さの)1次元系列(音声でも、気温の変化でも何でもよい)を3つ用意し、プロットせよ。それらの系列の類似度を適切な計算によって求め、考察せよ。

統計学の相関係数は、同じ長さのベクトル x, y の x_i, y_i が組になっていると見なしたときに、次のように計算される。

$$\frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad (15)$$

ただし、 \bar{x}, \bar{y} はそれぞれ x, y の(相加)平均である。

練習 51 下線部を埋めよ。

式(15)は、ベクトル (3) と (4) がなす角を θ とすると (5) と等しくなる。

14.2 相互相関関数

長さを同じにそろえた系列どうしであれば、内積やユークリッド距離を簡単に計算できる。しかし、実際に信号が似ているかどうかを調べたいときには、長さがまちまちであったり、一部だけが似ているということが多い。例えば、ある CM が、あるテレビ曲の番組で流されたかどうかを調べたいとする。すると、 x を CM (の音声)、 y をテレビ番組 (の音声) とすると y のどの部分に x (と似た信号) が含まれているかを調べる問題になる。

このようなときに利用できるのが相互相関関数 `xcorr` である。

```
>> y=[8 8 -3 4 -6 -10];
>> x=[8 -3 4 -6];
>> xc=xcorr(x,y)
xc =
-80.0000 -18.0000 10.0000 0.0000 125.0000 4.0000 26.0000 -16.0000 -48.0000 0.0000
0.0000
>> plot(xc)
```

この例では、短い x が長い y の 2 番目から 5 番目に含まれている。

この `xcorr` は、長さが異なるベクトルの場合には、長さをそろえてから、ずらしながら内積を計算している。`help` コマンドでわかるように、長さが異なる場合には、短いベクトルに 0 を付加する。

この場合、 x を $[8 -3 4 -6]$ から $[8 -3 4 -6 0 0]$ とする。

以後、ずらしながら内積を計算するまず、

```
8 8 -3 4 -6 -10
      8 -3 4 -6 0 0
```

とずらして対応をとる (-10 と 8 の対応がとられている)。重なっているところをベクトルとしてみると $[-10]$ と $[8]$ となる。

```
>> dot([-10],[8])
ans =
-80
```

であるので、`xcorr(x,y)` の 1 番目の要素は -80 となっている。

以後、1 つずつずらして同様に計算する。2 番目の要素を計算するときには、

```
8 8 -3 4 -6 -10
      8 -3 4 -6 0 0
```

となる。したがって、-18 となる。

このように計算すると、(6) 番目の要素の計算のときに、 y の x と一致した場所と x の内積が計算されることがわかる。

このような計算方法なので、`xcorr(x,y)` の計算結果の要素数は、ベクトル x,y の長い方の要素数を M とすると、(7) となる。したがって、計算結果の (8) 番目が、先頭をあわせたときの内積の値となる。そのインデックスを 0 とすると、上記の例では、`xc` はインデックスが (9) のとき最大となる。内積が (10) のときに、2 つのベクトルのなす角度が小さくなるため、`xc` の最大値を調べると、 x と y がどれくらい似ているか、どれだけずらしたときに最も似ているかの両方がわかる。

練習 52 下線部 (6) から (10) を埋めよ。

内積を計算しているため、(もちろん)`xcorr` は全く同一の系列でなくても、似ている部分があるかどうか、どこにあるかを調べることができる。

練習 53 適当な信号を 2 つ用意し、相互相関を計算し、どの部分が似ているかを調べよ。長さが長いとメモリ不足で相互相関が計算できないことがあるので、適当な長さの信号で試すこと。

15 自己相関

ある関数 x が周期関数であるということは、周期が T 秒だとすると、

$$x(t) = x(t + T) \quad (16)$$

とあらわされる。離散データとしては、 N 点が周期だとすると

$$x[n] = x[n + N] \quad (17)$$

となる。

この式を、部分が似ている、という視点で見ると、周期関数はその信号自身のある部分と、いくつかずらした部分と同じである、ということがわかる。

この考えに基づくと、前節の相互相関を同じベクトルに対して計算すると、いくつずらせば似るかを求められることがわかる。

ソースコード 15: xcorr の使用例

```
1 >> fs=100;
2 >> t=0:1/fs:1;
3 >> y=sin(2*pi*40*t);
4 >> xc=xcorr(y, 'coeff');
5 >> z=-----
6 >> plot((z:z+20)-z,xc(z:z+20))
```

n

MATLAB では、`xcorr(y)` は `xcorr(y,y)` と同じである。したがって、ソースコード 15 では、 y の長さは (11) なので、`xc` の (12) 番目の値が先頭を揃えた相互相関値となる。自分自身との比較なので、先頭を揃えたときが必ず最大値となる。したがって、その値が 1 になるように、最大値で全体を正規化することが多い。ソースコード 15 の 4 行目の 'coeff' というオプションは、そのような正規化を行うことを示す。また、同じ信号を前後にずらすだけなので、`xc` の値は、(12) 番目の値を中心として線対称の関係になる。

したがって、(12) 番目の値を 0 としてインデックスを設定することが一般的である。ソースコード 15 の 6 行目では、そのようなインデックスにして 0 点ずらしたとき (先頭をそろえたとき) から 20 点ずらしたときまでをプロットしている。(横軸はずらし幅の点数である)

このプロットで、0 点ずらしたときのピークの次に十分に大きいピークがあると、それが周期である可能性が高い。

練習 54 下線部 (11)、(12) を埋めよ。また、ソースコード 15 の下線部を埋めよ。

練習 55 ソースコード 15 のプロットの横軸の単位が時間になるようにしてプロットせよ。そのプロット結果から、周期が何秒なのかを読み取れ。

このような自分自身に対して相互相関を計算する関数をとくに自己相関関数とよぶ。

練習 56 周期的であると思う信号に対し自己相関を計算し、その結果のプロットを観察して、周波数を推定せよ。

ソースコード 16: 白色雑音の自己相関

```
1 >> fs=100;
2 >> t=0:1/fs:1;
3 >> r=-1+2*rand(size(t));
4 >> plot(r)
5 >> xc=xcorr(r, 'coeff');
6 >> plot(xc)
```

ソースコード 16 は、白色雑音に対して自己相関を計算する例である。雑音は周期的ではない信号であるため、ずらし幅が 0 以外の部分では、ピークがあったとしても、小さい値になっている。

練習 57 ソースコード 16 の `fs` を例えば 8000 などどんどん大きな値にして実行してみよ。大きくしたときにも `xc` をプロットし、どのようになるかを観察せよ。観察結果について考察せよ。

16 時間波形のフレーム処理

単語や文章をしゃべった音声は、イントネーションやアクセントに起因して、周期が刻一刻と変化する。このようなときには、8章で述べたフレーム処理をするのがよい。

MATLAB の特徴を用いると、`for` ループを利用せずに音声をフレーム分割できる。

ソースコード 17: 行列によるインデックス指定

```
1 >> x=(1:9)*2
2
3 x =
4
5     2     4     6     8    10    12    14    16    18
6
7 >> ind=[1 3 5 7; 2 4 6 8; 3 5 7 9]
8
9 ind =
10
11     1     3     5     7
12     2     4     6     8
13     3     5     7     9
14
15 >> x(ind)
16
17 ans =
18
19     2     6    10    14
20     4     8    12    16
21     6    10    14    18
22
23 >>
```

`x` は偶数の行ベクトルである。7行目で3行4列の行列 `ind` を設定している。15行目では、この `ind` を使って、ベクトル `x` にアクセスしている。すると、返り値は、`ind` と同じ大きさの行列になり、`ind` の値に対応した値が代入される。

よく見ればわかるように、`ind` を用いると、フレーム幅 3、フレームシフト(ずらし幅) 2 のフレームを生成できる。

この `ind` のような行列は、`repmat` という関数を利用して簡単に作ることができる。

```
>> W=3;
>> N=4;
>> SP=2;
>> repmat((1:W)',1,N)+repmat((0:(N-1))*SP,W,1)

ans =

     1     3     5     7
     2     4     6     8
     3     5     7     9
```

`repmat` がどのような関数であるか `help` コマンドで調べること。

練習 58 `repmat` を用いて、1次元の時系列データをフレーム幅 `W`、オーバーラップ幅(重ねる点数) `N`、のフレームに分割する MATLAB スクリプトを書け。ただし、列方向がフレームになるように(つまり、1列目が第1フレームになるように)せよ。

課題

以下の課題をパワーポイントファイルに作成し、授業支援システムのみ Navi の第 5 回目の「準備課題」のところにアップロードすること。パワーポイントファイルは、18 ポイント以上の大きさの文字を使って作成すること。締切は 10/17(日) 午後 8 時とします。自動で締め切ってしまいますので、十分に注意して下さい。音を生成・加工する課題については、音データもパワーポイントのスライドに貼り付けるようにすること。

1. 練習 47 を解答せよ。
2. 練習 48 を解答せよ。
3. 練習 52 を解答せよ。
4. 練習 56 を解答せよ。
5. 練習 57 を解答せよ。
6. この資料を読んで、わからなかった点があれば、わからなかった点が解消されるような回答が期待できるような具体的な質問を考えて書くこと。(これに関しては、スライドではなく、別のファイルに作成してアップロードせよ)