

数理科学特論 A ~ 画像数学 ~ 第 5 回

シリーズ 2 : 直交変換による画像圧縮

(2) 行列の直交変換, ユニタリー変換

前回は, 画像をベクトルで表し, ベクトルを直交行列で主成分に変換することで情報量を圧縮する考え方について説明しました. 今回は, 画像を行列で表現した場合に, ベクトルの行列による変換がどのような変換に対応するかを説明し, これに基づく直交変換・ユニタリー変換の考え方と基底画像について説明します.

行列の Kronecker 積と変換

前回は画像をベクトルで表して, ベクトルの直交変換について説明しました. しかし, デジタル画像は輝度を表す数値が 2 次元に配置されているものですから, 画像は行列で表すほうが自然です. そこで, 行列の変換について考えてみましょう.

前回, 行列 P によって画像を表すベクトル x をベクトル z に変換するという操作を考えました. つまり,

$$z = Px \quad (1)$$

という変換です. このとき, 列ベクトル x と z は m^2 個の要素でできているとし, P は $m^2 \times m^2$ 行列であるとします.

ここで, 画像を表すベクトル x を画像を表す行列に書き換えてみます. 列ベクトル x を m 個の m 要素の列ベクトル $x_1, \dots, x_j, \dots, x_m$ に分けます. すなわち

$$x = \begin{pmatrix} x_1 \\ \vdots \\ x_j \\ \vdots \\ x_m \end{pmatrix} \quad (2)$$

とします. 例えば, 9 要素からなる列ベクトルを 3 要素の列ベクトル 3 個に分けるということです. 分けられた各列ベクトルを行方向に並べ替えて, $m \times m$ 行列 X を得ます. すなわち,

$$X = \begin{pmatrix} x_1 & \cdots & x_j & \cdots & x_m \end{pmatrix} \quad (3)$$

とします. 同様にして, ベクトル z から行列 Z を得ます.

さて, $m^2 \times m^2$ 行列 P が, 次のような 2 つの $m \times m$ 行列 C と R を使って, 次のように表されるとします.

$$C = \begin{pmatrix} c_{11} & \cdots & c_{1m} \\ \vdots & \ddots & \vdots \\ c_{m1} & \cdots & c_{mm} \end{pmatrix}, R = \begin{pmatrix} r_{11} & \cdots & r_{1m} \\ \vdots & \ddots & \vdots \\ r_{m1} & \cdots & r_{mm} \end{pmatrix} \quad (4)$$

$$P = \begin{pmatrix} r_{11}c_{11} \cdots r_{11}c_{1m} & r_{1m}c_{11} \cdots r_{1m}c_{1m} \\ \vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\ r_{11}c_{m1} \cdots r_{11}c_{mm} & r_{1m}c_{m1} \cdots r_{1m}c_{mm} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ r_{m1}c_{11} \cdots r_{m1}c_{1m} & r_{mm}c_{11} \cdots r_{mm}c_{1m} \\ \vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\ r_{m1}c_{m1} \cdots r_{m1}c_{mm} & r_{mm}c_{m1} \cdots r_{mm}c_{mm} \end{pmatrix} \quad (5)$$

つまり，行列 P は，行列 R の各要素に行列 C を「貼り付け」たものになっています．この関係を

$$P = R \otimes C \quad (6)$$

で表し，「 P が R と C の Kronecker 積 (Kronecker product) で表される」といいます．

これらの表現を使うと，ベクトル x の行列 P による変換は

$$Z = CXR' \quad (7)$$

と，行列 X の行列 C, R による変換で表されます．

これは，以下のように証明されます．ベクトル x, z の内部にある（縦にならんでいる） j 番目のベクトル（それぞれ x_j, z_j ）の i 番目の要素をそれぞれ x_{ij}, z_{ij} とすると，(1)(5) 式から

$$z_{ij} = \begin{pmatrix} r_{j1}c_{i1} \cdots r_{j1}c_{im} \cdots r_{jk}c_{i1} \cdots r_{jk}c_{im} \cdots r_{jm}c_{i1} \cdots r_{jm}c_{im} \end{pmatrix} \begin{pmatrix} x_{11} \\ \vdots \\ x_{m1} \\ \vdots \\ x_{1k} \\ \vdots \\ x_{mk} \\ \vdots \\ x_{1m} \\ \vdots \\ x_{mm} \end{pmatrix} \quad (8)$$

$$= \sum_{k=1}^m r_{jk} \sum_{l=1}^m c_{il} x_{lk}$$

と表されます（ k はベクトル x の内部にあるベクトルの番号， l は内部にある k 番のベクトル，つまり x_k の要素の番号）．一方， z_{ij} は行列 Z の i 行 j 列の要素ですから，(7) 式から

$$z_{ij} = (CX)_{i\bar{j}} R'_{j\bar{j}}$$

$$= \begin{pmatrix} \sum_{l=1}^m c_{il} x_{l1} \cdots \sum_{l=1}^m c_{il} x_{lk} \cdots \sum_{l=1}^m c_{il} x_{lm} \end{pmatrix} \begin{pmatrix} r_{j1} \\ \vdots \\ r_{jk} \\ \vdots \\ r_{jm} \end{pmatrix} \quad (9)$$

$$= \sum_{k=1}^m r_{jk} \sum_{l=1}^m c_{il} x_{lk}$$

となり，確かに (1) 式と (7) 式が同等であることがわかります．

さて， P の 2 つの列の内積は

$$\begin{aligned}
& \sum_{i=1}^m r_{1k}c_{il} \cdot r_{1k'}c_{il'} + \dots + \sum_{i=1}^m r_{jk}c_{il} \cdot r_{jk'}c_{il'} + \dots + \sum_{i=1}^m r_{mk}c_{il} \cdot r_{mk'}c_{il'} \\
= & \sum_{j=1}^m \sum_{i=1}^m r_{jk}c_{il} \cdot r_{jk'}c_{il'} \\
= & \sum_{j=1}^m r_{jk}r_{jk'} \sum_{i=1}^m c_{il}c_{il'}
\end{aligned} \tag{10}$$

となります。P が正規直交行列であるとすると、 $k = k', l = l'$ のときこの値は 1，その他の時は 0 となります。 (10) 式の前の $\sum_{i=1}^m r_{jk}c_{il} \cdot r_{jk'}c_{il'}$ は行列 R の k 列と k' 列の積，後の $\sum_{i=1}^m c_{il}c_{il'}$ は行列 C の l 列と l' 列の積ですから，R と C をいずれも正規直交行列とすることで P を正規直交行列とすることができます。

以上から，画像を表すベクトル x を正規直交行列 P で変換する演算は，ベクトル x を行列 X で表した場合，P が正規直交行列 C と R の Kronecker 積で表されるならば (7) 式のように表されることがわかりました。ところで，(7) 式を見ると，行列 C は行列 X の列に，行列 R は行列 X の行に作用していることがわかります*)。つまり，「P が C と R の Kronecker 積で表される」とは，「P の作用を画像 X の列に対する作用と行に対する作用に分解できる」ことを意味しています。このことを，変換 P が分離可能 (separable) であるといいます。

行列の直交変換，ユニタリー変換

前回説明したように，画像の直交変換は，画像を「重要な成分」と「重要でない成分」に分けることを目的としています。そこで，今回は (1) 式の変換でベクトル z の下の方の要素をほとんど 0 にしてしまう方法を説明しました。一方今回説明した (7) 式の表現では，Z が対角行列になるように行列 C, R を選ぶ方法が知られており，これを特異値分解 (singular value decomposition, SVD) といいます。 $m \times m$ 行列 Z が対角行列ならば，0 でない要素は m 個だけなので，前回と同様の情報圧縮が行われていると言えます。

しかし，このような行列 C, R は，それぞれの画像 X 1 つ 1 つに対して別々に選ばなければなりません。したがって，「いろいろな画像に一定の変換を施して，少ない情報量で画像を伝達する」という目的は特異値分解では果たすことができません。前回説明した KL 変換でも，対象とする画像群の分散共分散行列がわからなければならぬので，この問題は同じです。

そこで，行列 C, R を画像に依存せずに選んで，変換後の画像 Z を，値 0 の要素（あるいはほとんど 0 とみなせる要素）が，たいていの画像について「なるべく多く」なるようにします。この際，画像の行と列について異なる取り扱いをする理由は通常ないので， $C = R$ とします。すると (7) 式は

$$Z = RXR' \tag{11}$$

となります。R は正規直交行列ですから $RR' = I$ なので，逆変換は

$$X = R'ZR \tag{12}$$

となります。このような変換を画像の直交変換 (orthogonal transformation) といいます。また，R の要素に複素数を考えるときは， $RR'^* = I$ となるような行列 R を用います。記号 * は共役行列（各要素の複素共役をとった行列）を意味します。このような行列をユニタリー行列 (unitary matrix) といい，このとき (11)(12) の変換をユニタリー変換 (unitary transformation) といいます。

基底画像

ところで，変換後の画像 Z を，「要素のうち 1 つだけが 0 でない行列 m^2 個の和」，すなわち

*) ここで文字 C, R を選んでいるのは，C は列 (column) の頭文字，R は行 (row) の頭文字だからです。

$$Z = \begin{pmatrix} z_{11} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} + \begin{pmatrix} 0 & z_{12} & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} + \cdots + \begin{pmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & z_{mm} \end{pmatrix} \quad (13)$$

のように分解すると考えます。このとき、行列 R を行ベクトルを使って

$$\mathbf{r}'_j = (r_{j1} \cdots r_{jk} \cdots r_{jm}), R = \begin{pmatrix} \mathbf{r}'_1 \\ \vdots \\ \mathbf{r}'_j \\ \vdots \\ \mathbf{r}'_m \end{pmatrix} \quad (14)$$

と表すと、(12) 式は

$$\begin{aligned} X &= R' \begin{pmatrix} z_{11} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} R + R' \begin{pmatrix} 0 & z_{12} & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} R + \cdots + R' \begin{pmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & z_{mm} \end{pmatrix} R \\ &= R' z_{11} \begin{pmatrix} \mathbf{r}'_1 \\ \mathbf{0}' \\ \vdots \\ \mathbf{0}' \end{pmatrix} + R' z_{12} \begin{pmatrix} \mathbf{r}'_2 \\ \mathbf{0}' \\ \vdots \\ \mathbf{0}' \end{pmatrix} + \cdots + R' z_{mm} \begin{pmatrix} \mathbf{0}' \\ \vdots \\ \mathbf{0}' \\ \mathbf{r}'_m \end{pmatrix} \\ &= z_{11} \mathbf{r}'_1 \mathbf{r}'_1 + z_{12} \mathbf{r}'_1 \mathbf{r}'_2 + \cdots + z_{mm} \mathbf{r}'_m \mathbf{r}'_m \\ &= \sum_{i=1}^m \sum_{j=1}^m z_{ij} \mathbf{r}'_i \mathbf{r}'_j \end{aligned} \quad (15)$$

のように、 R の行 (R' の列) 同士の外積でできる行列に分解されます。これらの行列を基底画像 (basis image) といいます。このように書くと、変換後の画像 (行列) の各要素 z_{ij} は各々の基底画像の係数と考えることができます。すなわち、「元の画像 X は、 (ij) 番の基底画像を z_{ij} 倍して足しあわせたもの」と考えることができます。そこで、送信側と受信側が共通の基底画像を事前に知っているとして、もしも m^2 個ある z_{ij} のうちいくつか以外を 0 とみなせるならば、そのいくつかの z_{ij} だけを伝えることで、情報を圧縮して画像を伝達することができます。このとき、画像 X は 0 でない z_{ij} に対応する基底画像だけでほとんど表現されていることになります。

ここまでの説明で、基底画像を「正弦波を表す指数関数」に置き換えると、第 1 シリーズで説明した離散フーリエ変換の説明と同じであることに気がつくと思います。2次元離散フーリエ変換はユニタリー変換の 1 つとして表現され、基底画像にはいろいろな細かさの縦横の波の組み合わせが現れます。もしも画像 X に細かい波の成分があまり含まれていないならば、細かい波を表す基底画像は用いなくてもよく、それに対応する z_{ij} は伝えなくてもよい、ということになります。

これまでに、このような方法による画像圧縮のための種々のユニタリー変換が考えられています。このシリーズの最終回となる次回は、ユニタリー変換としての離散フーリエ変換、および JPEG 方式の画像圧縮で用いられる離散コサイン変換について説明します。

今日の参考文献

M. Petrou and P. Bosdogianni, *Image Processing The Fundamentals* (既出)

A. K. Jain, *Fundamentals of Digital Image Processing* (既出)