@物理のかぎプロジェクト

#執筆中/行列式の導出/ソース

Author: pulsar
Date: 執筆中

行列式の導出

行列式の定義を見ると，どうしてこのような式を考え付いたのか想像しにくいですね．行列式を使わずに連立１次方程式を解いて，行列式の導出を試みましょう．

３元連立１次方程式

一般の場合は式が複雑で考えにくいので，まず

$$
\begin{aligned}
a_1x + b_1y + c_1z &= d_1 \\
a_2x + b_2y + c_2z &= d_2 \\
a_3x + b_3y + c_3z &= d_3
\end{aligned}
$$

について考えましょう．|9dd4e461268c8034f5c8564e155c67a6| を求めるために

$$a_ix + b_iy + c_iz = d_i$$

の辺々に $\pm b_jc_k(i \neq j \neq k \neq i)$ をかけた

$$\pm(a_ib_jc_kx + b_ib_jc_ky + c_ib_jc_kz) = \pm d_ib_jc_k$$

に対して，例えば

$$
\begin{aligned}
+a_1b_2c_3x + b_1b_2c_3y + c_1b_2c_3z &= +d_1b_2c_3 \\
-a_2b_1c_3x - b_2b_1c_3y - c_2b_1c_3z &= -d_2b_1c_3
\end{aligned}
$$

を辺々加算すると |415290769594460e2e485922904f345d| の係数が

$$(b_1b_2 - b_2b_1)c_3 = 0$$

となります．また,

$$
\begin{aligned}
+a_1b_2c_3x + b_1b_2c_3y + c_1b_2c_3z &= +d_1b_2c_3 \\
-a_3b_2c_1x - b_3b_2c_1y - c_3b_2c_1z &= -d_3b_2c_1
\end{aligned}
$$

を加算すると |fbade9e36a3f36d3d676c1b808451dd7| の係数が 0 になります．|415290769594460e2e485922904f345d| の係数には |367d0e931a088df0a8e7addc945fa7eb| が, |fbade9e36a3f36d3d676c1b808451dd7| の係数には |009602b61b3aa17c7e249bdb3864d338| が含まれているのがポイントで, |81033fe7e7162fdaa29fd8b3525a25b4| ならば

Inline substitution_reference start-string without end-string.

$$
(s_{ijk}b_ib_j - s_{jik}b_jb_i)c_ky = 0
$$

|7008774ba2a3c07021d61c54294cb0ef| ならば

Inline substitution_reference start-string without end-string.

$$
(s_{ijk}c_ic_k - s_{kji}c_kc_i)b_jz = 0
$$

となるので, |39ea6ef2598040e5f573f664f028a80b| を初期値として符号 |f8d6856ce7090fa6159170fbc855477e| を |2acdabd94f93ff7d4c45f49f5bbea51e| によって順次定めると

$$
\begin{aligned}
+a_1b_2c_3x + b_1b_2c_3y + c_1b_2c_3z &= +d_1b_2c_3 \\
-a_1b_3c_2x - b_1b_3c_2y - c_1b_3c_2z &= -d_1b_3c_2 \\
+a_2b_3c_1x + b_2b_3c_1y + c_2b_3c_1z &= +d_2b_3c_1 \\
-a_2b_1c_3x - b_2b_1c_3y - c_2b_1c_3z &= -d_2b_1c_3 \\
+a_3b_1c_2x + b_3b_1c_2y + c_3b_1c_2z &= +d_3b_1c_2 \\
-a_3b_2c_1x - b_3b_2c_1y - c_3b_2c_1z &= -d_3b_2c_1
\end{aligned}
$$

が得られ，総和をとると |415290769594460e2e485922904f345d|，|fbade9e36a3f36d3d676c1b808451dd7| の係数がいずれも 0 になることが分かります.

置換による表現

集合 |ac53d6834f53864f6cce98789b19b970| に対する 1 対 1 写像を置換といい, とくに |ac53d6834f53864f6cce98789b1 の任意の 2 数だけを交換する置換を互換といいます．|865c0c0b4ab0e063e5caa3387c1a8741| と |363b122c528f54df4a0446b6bab05515| を交換する互換 |a2ab7d71a0f07f388ff823293c147d21| は

Inline substitution_reference start-string without end-string.

Inline substitution_reference start-string without end-string.

$$
\sigma(i) = j, \quad \sigma(j) = i, \quad \sigma(k) = k (k \neq i, j)
$$

ですが，これを |5e33e4b364139e91328f66f0e78efc2c| とかきます．任意の置換は互換の繰り返し（合成写像）で表現できます．表現の仕方はいろいろありますが，置換を表現するのに必要な互換の数は偶数か奇数かは変わりません．互換の数が偶数の置換を偶置換，奇数の置換を奇置換といい，置換の符号を

$$\mathrm{sgn}(\text{偶置換}) = 1,\quad \mathrm{sgn}(\text{奇置換}) = -1$$

で定めます．|f8d6856ce7090fa6159170fbc855477e| の |e11e7dc346a4f3f8196cc186c06f11d1| は互いに異なるので，置換 |a2ab7d71a0f07f388ff823293c147d21| を用いて

$$i = \sigma(1), j = \sigma(2), k = \sigma(3)$$

と表現でき，置換を用いると |7b8b965ad4bca0e41ab51de7b31363a1| 元連立 1 次方程式への拡張が容易になります．

一般化準備として，まず

$$s_{ijk} b_j c_k (a_i x + b_i y + c_i z) = s_{ijk} d_i b_j c_k$$

を置換を用いて書き換えましょう．|80fed756c8358e6a19850beb8178f9bd| とし

$$\begin{aligned} i = \sigma(1), j = \sigma(2), k = \sigma(3) \\ a_i = a_{i1}, b_i = a_{i2}, c_i = a_{i3} \\ x = x_1, y = x_2, z = x_3 \end{aligned}$$

を代入した

$$s_\sigma a_{\sigma(2)2} a_{\sigma(3)3} \sum_{k=1}^{3} a_{\sigma(1)k} x_k = s_\sigma d_{\sigma(1)} a_{\sigma(2)2} a_{\sigma(3)3}$$

が |aa687da0086c1ea060a8838e24611319| を求める式であることに注意．|8732099f74d777a67257cb2f04ead3d8| を求めるときの式は

$$s_\sigma a_{\sigma(2)1} a_{\sigma(3)3} \sum_{k=1}^{3} a_{\sigma(1)k} x_k = s_\sigma d_{\sigma(1)} a_{\sigma(2)1} a_{\sigma(3)3}$$

あるいは |a2ab7d71a0f07f388ff823293c147d21| を変更した

$$s_\sigma a_{\sigma(1)1} a_{\sigma(3)3} \sum_{k=1}^{3} a_{\sigma(2)k} x_k = s_\sigma d_{\sigma(2)} a_{\sigma(1)1} a_{\sigma(3)3}$$

であり，|1f89889020cdc84d9e1c35237cb62f65| を求める式は

$$s_\sigma \sum_{k=1}^{3} a_{\sigma(j)k} x_k \prod_{I \neq j} a_{\sigma(i)i} = s_\sigma d_{\sigma(j)} \prod_{i \neq j} a_{\sigma(i)i}$$

です．上式の 3 を |7b8b965ad4bca0e41ab51de7b31363a1| で置換し, |a2ab7d71a0f07f388ff823293c147d21| の定義域を |baf9bcdf3df7765be566d31a4efa8b3d| と考えれば，そのまま一般の場合に適用できます．

一般化

3 を |7b8b965ad4bca0e41ab51de7b31363a1| で置換し，|a2ab7d71a0f07f388ff823293c147d21| の定義域を |baf9bcdf3df7765be566d31a4efa8b3d| と考えても |1f89889020cdc84d9e1c35237cb62f65| を同じ式

で求められることを確かめましょう．

$$\sum_{k=1}^{n} a_{\sigma(j)k} x_k s_\sigma \prod_{i \neq j} a_{\sigma(i)i} = d_{\sigma(j)} s_\sigma \prod_{i \neq j} a_{\sigma(i)i}$$

の |a2ab7d71a0f07f388ff823293c147d21| についての総和をとると，|945315e9d656ad23442dd758248b758e| の係数は

$$\sum_{\sigma} s_\sigma a_{\sigma(j)k} a_{\sigma(k)k} \prod_{i \neq j,k} a_{\sigma(i)i} = 0$$

となります．ここで |82cb38cfb7b8f079dda70c6a96f37479| は |2468f06b771fbc7d1f1af1eea1aed3db|, |c0b0371dad9f5aa0cb29a960a14d491f| は |1ef7668554b437666dc6bca399e6ff14| を意味します．上式が成立することは |e4f7b01030892a5ebb9a21e5fded02be| である任意の置換 |88207bc086c9d879c22807479d29a9e2| に対して置換 |2cbc2f0a6d893fbbd534422e917a3281| が存在して，

Inline substitution_reference start-string without end-string.

Inline substitution_reference start-string without end-string.

Inline substitution_reference start-string without end-string.

$$s_{\sigma'} a_{\sigma'(j)k} a_{\sigma'(k)k} + s_{\sigma''} a_{\sigma''(j)k} a_{\sigma''(k)k} = 0$$
$$a_{\sigma'(i)i} = a_{\sigma''(i)i} (i \neq j, k)$$

となることで証明できます．

補遺

(1) 発見的に考えるには対象を簡単化して見易い記号を使うこと．最初から

$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n = d_i$$

で考えようとすると無用な複雑さで思考が妨げられます．

(2)「3元連立1次方程式」では |6ad43a65711267334c0d010419163143| に $\pm b_j c_k$ を天下り的にかけましたが，

$$a_1x + b_1y + c_1z = d_1$$
$$a_2x + b_2y + c_2z = d_2$$

から |fbade9e36a3f36d3d676c1b808451dd7| を消去すると

$$(c_1a_2?c_2a_1)x + (c_1b_2?c_2b_1)y = c_1d_2?c_2d_1$$

が得られ，同様に

$$(c_2a_3?c_3a_2)x + (c_2b_3?c_3b_2)y = c_2d_3?c_3d_2$$
$$(c_3a_1?c_1a_3)x + (c_3b_1?c_1b_3)y = c_3d_1?c_1d_3$$

も成立するので，|415290769594460e2e485922904f345d| の係数に注目して

$$
\begin{array}{rclll}
b_3(c_1b_2?c_2b_1) & = & b_2b_3c_1 & -b_3b_1c_2 & \\
b_1(c_2b_3?c_3b_2) & = & & b_3b_1c_2 & -b_1b_2c_3 \\
b_2(c_3b_1?c_1b_3) & = & -b_2b_3c_1 & & +b_1b_2c_3
\end{array}
$$

から，加重加算によって |415290769594460e2e485922904f345d| の係数を にできることが分かります．行列式で表すと

$$
-b_1\begin{vmatrix} b_2 & c_2 \\ b_3 & c_3 \end{vmatrix} + b_2\begin{vmatrix} b_1 & c_1 \\ b_3 & c_3 \end{vmatrix} - b_3\begin{vmatrix} b_1 & c_1 \\ b_2 & c_2 \end{vmatrix} = 0
$$

です．

(3) 連立 1 次方程式

$$
\sum_{k=1}^{n} a_{ik}x_k = d_i
$$

の解 |1f89889020cdc84d9e1c35237cb62f65| は

|16868711d2f80fc5c4b56ed9721a2224| を |e9e0a5801a1a99ca1d3b024c594095bc| 要素とする |7b8b965ad4bca0e41ab51de7b31363a1| 次正方行列 |7fc56270e7a70fa81a5935b72eacbe29| の行列式は

Inline substitution_reference start-string without end-string.

$$
|A| = \sum_{\sigma} \mathrm{sgn}(\sigma) \prod_{i=1}^{n} a_{\sigma(i)i}
$$

で定義されるので, |1f89889020cdc84d9e1c35237cb62f65| の係数が |16868711d2f80fc5c4b56ed9721a2224| を |dadd98b275f56f5ef19c1bc63ba2dccb| 要素とする |7b8b965ad4bca0e41ab51de7b31363a1| 次正方行列 |7fc56270e7a70fa81a5935b72eacbe29| の行列式であり, 上式右辺は行列 |7fc56270e7a70fa81a5935b72eacbe29| の |abef081fd754a0a09d3b9c4dae967b28| 要素を |d247f594c78d0d2be10fc6d82512cc4e| で置換した行列の行列式になっていることを確かめられます．

あとがき