A TECHNOLOGY REPORT

SUBJECT

A LAW OF BLOOD TYPE

JOINT STUDY

JEY

ACCHONBURIKE UNIV.
UKYOKYO LABORATORY

PINOKO

2001.2.6

目 次

第1章　INTRODUCTION

1.1　JEY's Doubt

1. 　血液型にはＡ,Ｂ,Ｏ,ＡＢ型の４種類存在するが、今後日本の血液型の割合はどう変化していくのか？(淘汰される血液型があるかも？)
2. 世界の始めに同じＡ,Ｂ,Ｏ,ＡＢ型の血液型が同じ割合だけ存在したら現在の血液型はどうなるか？
3. また、その割合は現実の世界の血液型の割合と一致するか。
4. 世界の始めの血液型の割合は何か？
5. アダムとイブの血液型とその子供たちの血液型の割合は？
6. さらに、それぞれの世代で結婚する人の血液型の期待値は？
7. もし血液型因子が４種類以上あったらどうなるか。実際あったかもしれん。

について考えたレポートである。ちなみに日本人の血液型の割合はＡ型３８．１％、Ｂ型２１．８％、Ｏ型３０．７％、ＡＢ型９．４％だそうだ。

1.2　Presupposition

1. 血液型は血液型因子 (a,b,o) から決まっているものとし、以下の９種類で考える。

$$\begin{array}{cc} & \begin{array}{ccc} a & b & o \end{array} \\ \begin{array}{c} a \\ b \\ o \end{array} & \begin{pmatrix} aa & ab & ao \\ ba & bb & bo \\ oa & ob & oo \end{pmatrix} \end{array}$$

2. 以下血液因子を $a = 1$,$b = 2$,$o = 3$ で表す。
3. この血液型の割合を表した 3×3 の行列をＰ (Parent) 行列とする。Ｐ行列の成分を $P_{aa}, P_{ab}, \ldots, P_{oo}$ で表す。

4. そして、その親から生まれる子供の血液型の割合を親と同様にＣ (Child) 行列とする。Ｃ行列の成分を $C_{11}, C_{12}, \ldots, C_{33}$ で表す。

5. 子供の子供、つまり孫の血液型の割合を Ｃ′ 行列とるす。

6. $\sum P_{ij} = 1, \sum C_{ij} = 1, \sum C'_{ij} = 1$ が成立する。

7. Ｐ行列、Ｃ行列、Ｃ′ 行列は対象行列とする。

8. 男女の血液型の割合は同じであるとする。

9. それぞれの血液型が結婚する確率は同様に確からしいとする。

10. 数式　Ｐ＊Ｐ＝Ｃ（＊はある演算）が成り立つ。

第2章　Blood Type Operation

演算＊はどのようなものかを考える。

2.1　Conbination

まず組合せの種類を考えてみる。
aa 型と aa 型の親からは aa,aa,aa,aa の 4 種類。全部同じ。
ab 型と aa 型の親からは aa,aa,ba,ba の 4 種類。
ao 型と aa 型の親からは aa,aa,oa,oa の 4 種類。
。。。。
というように（書くのがめんどくさい）となる。親の組み合わせは $9 \times 9 = 81$ 通り。その親から生まれる子供は $81 \times 4 = 324$ 通りとなる。そこで、この３２４通りの子供の血液型は上記 9 種類の血液型に分けたときにそれぞれどのような割合になってるのだろうか。C 言語で初めてプログラムを作ってみた。その結果すべて３６組。割合でいえば３６／３２４＝１／９だ。考えてみれば、因子からすべての組み合わせを考えるのだからあたりまえじゃないか。

2.2　Math Examination

演算*を数式で表すとどうなるのか考えてみる。aa 型の子どもは aa,ab,ao,ba,oa の親からしか生まれない。よって $5 \times 5 = 25$ 通りの親の組合せからしか生まれない。それぞれ何組生まれるか考えてみると以下の表になる。

$$
\begin{array}{c c}
 & \begin{array}{ccccc} aa & ab & ao & ba & oa \end{array} \\
\begin{array}{c} aa \\ ab \\ ao \\ ba \\ oa \end{array} &
\begin{pmatrix}
4 & 2 & 2 & 2 & 2 \\
2 & 1 & 1 & 1 & 1 \\
2 & 1 & 1 & 1 & 1 \\
2 & 1 & 1 & 1 & 1 \\
2 & 1 & 1 & 1 & 1
\end{pmatrix}
\end{array}
$$

$P_{ab}=P_{ba}$、$P_{ao}=P_{oa}$ だから (∵前提 1.2.4 より　P 行列は対象行列) 整理すると。

$$\begin{array}{c} \\ aa \\ ab \\ ao \end{array} \begin{array}{c} \begin{array}{ccc} aa & ab & ao \end{array} \\ \begin{pmatrix} 4 & 4 & 4 \\ 4 & 4 & 4 \\ 4 & 4 & 4 \end{pmatrix} \end{array}$$

となる。つまり子供の aa 型の割合は

$$C_{aa}=(P_{aa}+P_{ab}+P_{ao})^2 \tag{2.1}$$

となる。同様にして

$$C_{bb}=(P_{ba}+P_{bb}+P_{bo})^2 \tag{2.2}$$

$$C_{oo}=(P_{oa}+P_{ob}+P_{oo})^2 \tag{2.3}$$

になる。では、ao,ab,bo 型はどうなるのか。ab 型の子供について考える。ab 型の子供は aa,ab,ao,ba,bb,bo,oa,ob 型の親から生まれる。ここで、ab という (a と b の順番が大切) 組み合わせを行と列に当てはめて考えると以下の表になる。

$$\begin{array}{c} \\ aa \\ ab \\ ao \\ ba \\ oa \end{array} \begin{array}{c} \begin{array}{ccccc} ab & ba & bb & bo & ob \end{array} \\ \begin{pmatrix} 2 & 2 & 4 & 2 & 2 \\ 1 & 1 & 2 & 1 & 1 \\ 1 & 1 & 2 & 1 & 1 \\ 1 & 1 & 2 & 1 & 1 \\ 1 & 1 & 2 & 1 & 1 \end{pmatrix} \end{array}$$

整理すると。

$$\begin{array}{c} \\ aa \\ ab \\ ao \end{array} \begin{array}{c} \begin{array}{ccc} ba & bb & bo \end{array} \\ \begin{pmatrix} 4 & 4 & 4 \\ 4 & 4 & 4 \\ 4 & 4 & 4 \end{pmatrix} \end{array}$$

よって

$$C_{ab}=(P_{aa}+P_{ab}+P_{ao})(P_{ba}+P_{bb}+P_{bo}) \tag{2.4}$$

同様にして、

$$C_{ao}=(P_{aa}+P_{ab}+P_{ao})(P_{oa}+P_{ob}+P_{oo}) \tag{2.5}$$

$$C_{bo}=(P_{ba}+P_{bb}+P_{bo})(P_{oa}+P_{ob}+P_{oo}) \tag{2.6}$$

となる

2.3 General Equation

血液型演算子の一般解

$$C_{ij} = (P_{i1} + P_{i2} + P_{i3})(P_{j1} + P_{j2} + P_{j3}) \tag{2.7}$$

2.4 The Matrix Operation

一般解が解かったところで、行列演算を考えてみる。

$$\mathrm{P} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \mathrm{P} = \mathrm{C} \tag{2.8}$$

これが、血液型演算行列だあ !!　発見 !!

2.5 Pascal's triangle

これを考えてたらパスカルの三角形を思い出した。(二項定理っていうんだっけ）でも今回はパスカルの 3 角錐だなあ。ところで、どうして 3 角形なんだろう僕は 3 次元以上を考えたらパスカルの 4 角形とかパスカルの立方体とかいったほうがいいような気がする。でも、やっぱり 3 角かなあ。どおでもいいけど。

第3章　Blood Operarion Property

血液型演算子（勝手に名前つけた）の性質はどうなってるのか。コンピュータに計算させてみると最初の割合を血液因子の割合で決めるとずっと後の世代も同じになる。でも、最初の割合を変えると一回目の演算で割合が変わりその後は、ずっとその割合がつづく。うーん、不思議だ。

3.1　Aproch from general equation

ということでまずは、一般解からの考察。

$$P_{i1} + P_{i2} + P_{i3} = S_i \text{とする} \tag{3.1}$$

$$\begin{aligned} C_{ij} &= (P_{i1} + P_{i2} + P_{i3})(P_{j1} + P_{j2} + P_{j3}) \\ &= S_i S_j \end{aligned} \tag{3.2}$$

$$\begin{aligned} C'_{ij} &= (C_{i1} + C_{i2} + C_{i3})(C_{j1} + C_{j2} + C_{j3}) \\ &= (S_i S_1 + S_i S_2 + S_i S_3)(S_j S_1 + S_j S_2 + S_j S_3) \\ &= (S_i(S_1 + S_2 + S_3))(S_j(S_1 + S_2 + S_3)) \\ &= S_i S_j (S_1 + S_2 + S_3)^2 \\ &= S_i S_j (\because\ S_1 + S_2 + S_3 = 1) \\ &= C_{ij} \end{aligned} \tag{3.3}$$

証明終了。

3.2　Aproch from the Matrix Operation

次に行列からの考察。

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} = \mathrm{Q}\text{とおくと。} \tag{3.4}$$

$$\mathrm{C} = \mathrm{P\,Q\,P} \tag{3.5}$$

$$\mathrm{C}' = \mathrm{C\,Q\,C} = \mathrm{P\,Q\,P\,Q\,P\,Q\,P} \tag{3.6}$$

である。ここで、ＱＰＱに着目する。ぐちゃぐちゃ計算するとＱＰＱ＝Ｑであることがわかる（$\because \sum P_{ij} = 1$）よって

$$\mathrm{C}' = \mathrm{P\,Q\,P\,Q\,P\,Q\,P} = \mathrm{P\,Q\,P\,Q\,P} = \mathrm{P\,Q\,P} = \mathrm{C} \tag{3.7}$$

証明終了。

[余談]

ＱＰＱ＝ＱだからといってＱＰ＝Ｅではない。なぜならＱの逆行列 Q^{-1} は行列式 $|\,\mathrm{Q}\,| = 0$ より、存在しない。Ｑ行列の逆行列が存在するための必要十分条件はＱの行列式 $|\,\mathrm{Q}\,| \neq 0$ である。

血液型演算子による変換は、１回目の配合（２世代目）以下の血液型の割合は変化しないということが証明された。

第4章　A law of Hardy Weinberg

世の中にはハーディ・ワインベルグの法則というものが存在するということがわかった。ホームページ（http://www.primate.gr.jp/yasuda/6.html）にいろいろ書いてある。ひょっとして、今まで書いてきたことはこの法則の証明なのだろうか。難しくてよくわからんが、ちょっと違う気がする。なんてったって、生物学を何も知らないから、染色体のことを考えてない。男は母親依存だからなんとかかんとか、と書いてある。そんなこと考えてないからなあ。

4.1　A law of Hardy Weinberg

ハーディ・ワインベルグの法則を簡単に書いておく。
血液型Ａ，Ｂ，Ｏ，ＡＢの割合は因子の割合 a,b,o により以下のように表され

$$A = a^2 + 2ao, B = b^2 + 2ao, O = o^2, AB = 2ab \tag{4.1}$$

そして、この血液型の割合は安定している。というものである。

では、実際にやってみる。現在の日本の血液型はＡ型３８．１％、Ｂ型２１．８％、Ｏ型３０．７％、ＡＢ型９．４％だから。代入して計算すると、a=27.5％,b=17.1％,o=55.4％となり、因子の割合が決定される。確かに未知数３つに対し式が４つなので（正確にはそれぞれ足して１なので未知数２つに対し式が３つだが）普通なら矛盾するはず。でも、血液型は血液因子によって決まってるのだからこれはあたりまえだ。ハーディ・ワインベルグの法則は後者、安定しているということだ。安定しているかどうか考えてみる。a,b,o 因子の割合を a,b,o とすると。($a + b + o = 1$)

$$P_{ij} = ij \tag{4.2}$$

となる。この親から子供の血液型の割合を考えると。

$$\begin{aligned} C_{ij} &= (ia + ib + io)(ja + jb + jo)(\because \text{一般解より}) \\ &= iaja + iajb + ia + jo + ibja + ibjb + ibjo + ioja + iojb + iojo \\ &= ij(a^2 + b^2 + o^2 + 2ab + 2ao + 2ab) \\ &= ij(a + b + o)^2 \\ &= ij(\because\ a + b + o = 1) \end{aligned} \tag{4.3}$$

証明終了
となり、親の血液型の割合と子供の血液型の割合は変化しない。確かに安定しているといえる。でもハーディ・ワインベルグの法則ってこういう理解でいいのかどうか …

第5章　The blood type of the world at first

いろいろわかったところで jey の疑問の一つ、世界の最初の血液型の割合は？につて考えてみる。血液型因子で血液型は決まっているとして今まで考えてきた。そうであるならば、前章のように、因子の割合が決定すれば、それぞれの血液型の割合は一意的に決まってくる。しかし一般解を血液因子という条件で範囲を狭めてしまう。

5.1　Extend a law of Hardy Weinberg

因子の割合が決まってから、Ａ, Ｂ, Ｏ, ＡＢ型の割合が決まるのではなく、Ａ, Ｂ, Ｏ, ＡＢ型の割合が偶発的に決まってから因子の割合が決まったとすると、どうなるのか。実際Ａ, Ｂ, Ｏ, ＡＢ型の割合を適当に決めると、a,b,o 因子の割合は式が多すぎて矛盾する。しかし、一回配合すると一定の値に収束することから、その収束した割合は a,b,o 因子で表現できるかもしれない。かってに決めたＡ, Ｂ, Ｏ, ＡＢ型の割合から生まれる子供の血液型の割合は血液因子の割合を特定できるのか。一回配合することと、血液因子の割合が決まることは同値なのか。もしそうだとすると、Ａ, Ｂ, Ｏ, ＡＢ型の割合がなんであっても、一回配合すれば、a,b,o 因子の割合が決まることになる。

5.2　the relation between blood factor and general equation

血液因子と一般解の関係を考える。親の血液因子の割合を a_p, b_p, o_p で表す。子供の血液因子の割合を a_c, b_c, o_c で表す。親の関係は

$$\begin{pmatrix} P_{aa} & P_{ab} & P_{ao} \\ P_{ba} & P_{bb} & P_{bo} \\ P_{oa} & P_{ob} & P_{oo} \end{pmatrix} = \begin{array}{c|ccc} & a_p & b_p & o_p \\ \hline a_p & a_p^2 & a_p b_p & a_p o_p \\ b_p & b_p a_p & b_p^2 & b_p o_p \\ o_p & o_p a_p & o_p b_p & o_p^2 \end{array} \tag{5.1}$$

となる。親の血液型の割合 (P_{ij}) を適当に決めた場合 a_p, b_p, o_p は決定されない。関係式が多すぎて矛盾する。

次に子供の関係を考える。

$$P_{i1}+P_{i2}+P_{i3}=S_i \text{とする} \tag{5.2}$$

一般解から

$$\begin{aligned} C_{ij} &= (P_{i1}+P_{i2}+P_{i3})(P_{j1}+P_{j2}+P_{j3}) \\ &= S_iS_j \end{aligned} \tag{5.3}$$

だから

$$\begin{pmatrix} C_{aa} & C_{ab} & C_{ao} \\ C_{ba} & C_{bb} & C_{bo} \\ C_{oa} & C_{ob} & C_{oo} \end{pmatrix} = \begin{pmatrix} S_a^2 & S_aS_b & S_aS_o \\ S_bS_a & S_b^2 & S_bS_o \\ S_oS_a & S_oS_b & S_o^2 \end{pmatrix} = \begin{array}{c c} & \begin{array}{ccc} a_c & b_c & o_c \end{array} \\ \begin{array}{c} a_c \\ b_c \\ o_c \end{array} & \begin{pmatrix} a_c^2 & a_cb_c & a_co_c \\ b_ca_c & b_c^2 & b_co_c \\ o_ca_c & o_cb_c & o_c^2 \end{pmatrix} \end{array} \tag{5.4}$$

よって

$$a_c = S_a = P_{aa}+P_{ab}+P_{ao} \tag{5.5}$$

$$b_c = S_b = P_{ba}+P_{bb}+P_{bo} \tag{5.6}$$

$$o_c = S_c = P_{oa}+P_{ob}+P_{oo} \tag{5.7}$$

となり、血液因子は決定される。つまり、これは親の血液型の割合が何であっても子供の血液型は血液因子の割合で表現できる。言い方を変えると親の血液型が血液因子の割合で表現できなくても、その後の代では血液因子の割合が決まり、血液型の割合は血液因子の割合によって表現できる。

ここで、ＪＥＹの疑問、世界の始めの血液型は？の回答が得られる。

現在の血液型因子の割合がわかっていても、世界の始めの血液型の割合は特定できない。

第6章　Conclusion

＜結論＞

1. 今後の血液型の割合は変化しない。
2. 現在の血液因子の割合がわかっていても、世界の始めの血液型の割合は特定できず、無限に存在する。

これが、ＰＩＮＯＫＯとＪＥＹの法則である。実は自明なことだったりして。

第7章　C Program

今回初めてC言語で作ったプログラムを参考に書いておきます。でも、こんなの乗せるまでもないけど、一応念のため。

```
/*
   ## BLOOD TYPE ANLYZE PLOGRAM ##
   programing by PINOKO 2001/02/03
    (a=1:b=2:o=3)
p:parents ratio
c:child ratio (computer summention valu)
d:child ratio (math caluculation valu)
*/

#include<stdio.h>

main()
{
  double p[4][4],c[4][4],d[4][4];
  int i,j,k,l,n,m;

  n=0;

  p[1][1]=0.1905 ;p[1][2]=0.047 ;p[1][3]=0.09525;
  p[2][1]=0.0470 ;p[2][2]=0.109 ;p[2][3]=0.0545;
  p[3][1]=0.09525;p[3][2]=0.0545;p[3][3]=0.307;

  for(i=1;i<4;++i)
    { for(j=1;j<4;++j)
      d[i][j]=0.0;
    }

  printf("m=%d  ",m);
  printf("n=%d\n",n);
```

```
  printf("p(1,1)=%lf (%lf) "  ,p[1][1],d[1][1]);
  printf("p(1,2)=%lf (%lf) "  ,p[1][2],d[1][2]);
  printf("p(1,3)=%lf (%lf)\n" ,p[1][3],d[1][3]);
  printf("p(2,1)=%lf (%lf) "  ,p[2][1],d[2][1]);
  printf("p(2,2)=%lf (%lf) "  ,p[2][2],d[2][2]);
  printf("p(2,3)=%lf (%lf)\n" ,p[2][3],d[2][3]);
  printf("p(3,1)=%lf (%lf) "  ,p[3][1],d[3][1]);
  printf("p(3,2)=%lf (%lf) "  ,p[3][2],d[3][2]);
  printf("p(3,3)=%lf (%lf)\n" ,p[3][3],d[3][3]);

  printf("[A  type=%lf]  ",p[1][1]+p[1][3]+p[3][1]);
  printf("[AB type=%lf]  ",p[1][2]+p[2][1]);
  printf("[B  type=%lf]  ",p[2][2]+p[2][3]+p[3][2]);
  printf("[O  type=%lf]\n",p[3][3]);

  printf("A+B+AB+O=%lf\n",p[1][1]+p[1][2]+p[1][3]+
                          p[2][1]+p[2][2]+p[2][3]+
                          p[3][1]+p[3][2]+p[3][3]);
  printf("\n");

  for(m=1;m<3;++m)
    {
      printf("m=%d  ",m);

      c[1][1]=0;c[1][2]=0;c[1][3]=0;
      c[2][1]=0;c[2][2]=0;c[2][3]=0;
      c[3][1]=0;c[3][2]=0;c[3][3]=0;

      for(i=1;i<4;++i)
{ for(j=1;j<4;++j)
  { for(k=1;k<4;++k)
    { for(l=1;l<4;++l)
      {
n++;
c[i][k] += 0.25*p[i][j]*p[k][l];
c[i][l] += 0.25*p[i][j]*p[k][l];
c[j][k] += 0.25*p[i][j]*p[k][l];
c[j][l] += 0.25*p[i][j]*p[k][l];
      }
```

```
    }
  }
}

      for(i=1;i<4;++i)
{ for(j=1;j<4;++j)
  d[i][j]=(p[i][1]+p[i][2]+p[i][3])*(p[j][1]+p[j][2]+p[j][3]);
}

      printf("n=%d\n",n);

      printf("c(1,1)=%lf (%lf) "  ,c[1][1],d[1][1]);
      printf("c(1,2)=%lf (%lf) "  ,c[1][2],d[1][2]);
      printf("c(1,3)=%lf (%lf)\n" ,c[1][3],d[1][3]);
      printf("c(2,1)=%lf (%lf) "  ,c[2][1],d[2][1]);
      printf("c(2,2)=%lf (%lf) "  ,c[2][2],d[2][2]);
      printf("c(2,3)=%lf (%lf)\n" ,c[2][3],d[2][3]);
      printf("c(3,1)=%lf (%lf) "  ,c[3][1],d[3][1]);
      printf("c(3,2)=%lf (%lf) "  ,c[3][2],d[3][2]);
      printf("c(3,3)=%lf (%lf)\n" ,c[3][3],d[3][3]);

      printf("[A  type=%lf]  ",c[1][1]+c[1][3]+c[3][1]);
      printf("[AB type=%lf]  ",c[1][2]+c[2][1]);
      printf("[B  type=%lf]  ",c[2][2]+c[2][3]+c[3][2]);
      printf("[O  type=%lf]\n",c[3][3]);

      printf("A+B+AB+O=%lf\n",c[1][1]+c[1][2]+c[1][3]+
                              c[2][1]+c[2][2]+c[2][3]+
                              c[3][1]+c[3][2]+c[3][3]);
      printf("\n");

      p[1][1]=c[1][1];p[1][2]=c[1][2];p[1][3]=c[1][3];
      p[2][1]=c[2][1];p[2][2]=c[2][2];p[2][3]=c[2][3];
      p[3][1]=c[3][1];p[3][2]=c[3][2];p[3][3]=c[3][3];
    }
}
```

出力結果はこんな感じ。

```
[einstein c]$ ./a.out
m=0  n=0
p(1,1)=0.190500 (0.000000) p(1,2)=0.047000 (0.000000) p(1,3)=0.095250 (0.000000)
p(2,1)=0.047000 (0.000000) p(2,2)=0.109000 (0.000000) p(2,3)=0.054500 (0.000000)
p(3,1)=0.095250 (0.000000) p(3,2)=0.054500 (0.000000) p(3,3)=0.307000 (0.000000)
[A  type=0.381000]  [AB type=0.094000]  [B  type=0.218000]  [O  type=0.307000]
A+B+AB+O=1.000000

m=1  n=81
c(1,1)=0.110723 (0.110723) c(1,2)=0.070044 (0.070044) c(1,3)=0.151984 (0.151984)
c(2,1)=0.070044 (0.070044) c(2,2)=0.044310 (0.044310) c(2,3)=0.096146 (0.096146)
c(3,1)=0.151984 (0.151984) c(3,2)=0.096146 (0.096146) c(3,3)=0.208621 (0.208621)
[A  type=0.414690]  [AB type=0.140088]  [B  type=0.236602]  [O  type=0.208621]
A+B+AB+O=1.000000

m=2  n=162
c(1,1)=0.110723 (0.110723) c(1,2)=0.070044 (0.070044) c(1,3)=0.151984 (0.151984)
c(2,1)=0.070044 (0.070044) c(2,2)=0.044310 (0.044310) c(2,3)=0.096146 (0.096146)
c(3,1)=0.151984 (0.151984) c(3,2)=0.096146 (0.096146) c(3,3)=0.208621 (0.208621)
[A  type=0.414690]  [AB type=0.140088]  [B  type=0.236602]  [O  type=0.208621]
A+B+AB+O=1.000000
```