マルコフ基底と実験計画, 問題集その 2, 2009.09.15

1 計画 (design) と計画イデアル (design ideal)

k を体とする. ここでは主に $k = \mathbf{Q}$ として考える.

定義 1 k^n の相異なる点の有限集合を計画 (design) とよぶ.

問 1 次の実施計画の計画点を具体的に書け.

1. D_1 : 2^2-完全実施計画 (水準 $\{-1, 1\}$)
2. D_2 : 2^2-完全実施計画 (水準 $\{0, 1\}$)
3. D_3 : 3^3-完全実施計画 (水準 $\{-1, 0, 1\}$) の中で少なくとも 1 つの変数が 0 であるもの.
4. D_4 : 別名関係 $ACD = I$ をもつ 2^{4-1}-一部実施計画 (水準 $\{-1, 1\}$)
5. D_5 : 別名関係 $ABC^2 = BCD = I$ $(a + b + 2c \equiv 0 \pmod 3), b + c + d \equiv 0 \pmod 3))$ を持つ 3^{4-2}-一部実施計画 (水準 $\{-1, 0, 1\}$)
6. D_6 : 別名関係 $ABDE = ACDF = BCDG = I$ を持つ 2^{7-3}-一部実施計画 (水準 $\{-1, 1\}$)

例 1 (1) $D_1 = \{-1, 1\}^2 = \{(1, 1), (1, -1), (-1, 1), (-1, -1)\}$

参考文献 : [1, chap 3.1]

定義 2 計画 D に対して, D の任意の点を零点にもつ全ての多項式の集合

$$\mathbf{I}(D) = \{f(x) \in \mathbf{Q}[x] := \mathbf{Q}[x_1, \ldots, x_n] \mid f(a) = 0 \ (\forall a \in D)\}$$

を D の計画イデアル (design ideal) という.

1 点 $a = (a_1, \ldots, a_n) \in \mathbf{Q}^n$ のみを計画点としてもつ計画の計画イデアルは, 定義より,

$$\langle x_1 - a_1, \ldots, x_n - a_n \rangle \subset \mathbf{Q}[x]$$

であるので, m 点 $a_i = (a_{i1}, \ldots, a_{in})$ $(1 \leq i \leq m)$ からなる計画の計画イデアルは

$$\bigcap_{i=1}^{m} \langle x_1 - a_{i1}, \ldots, x_n - a_{in} \rangle$$

である. 従って, イデアルの交わりを計算すればよい.

例 2 $D_1 = \{-1, 1\}^2 = \{(1, 1), (1, -1), (-1, 1), (-1, -1)\}$ の計画イデアル

$$\begin{aligned}\mathbf{I}(D_1) &= \langle x - 1, y - 1 \rangle \cap \langle x - 1, y + 1 \rangle \cap \langle x + 1, y - 1 \rangle \cap \langle x + 1, y + 1 \rangle \\ &= \langle x^2 - 1, y^2 - 1 \rangle\end{aligned}$$

2 つのイデアル $I = \langle f_1, \dots, f_k \rangle$, $J = \langle g_1, \dots, g_\ell \rangle$ の交わりは

$$\langle tf_1, \dots, tf_k, (1-t)g_1, \dots, (1-t)g_\ell \rangle \cap k[x]$$

と計算できる [2, chap 4.3]. 例 2 の 1 つ目と 2 つ目のイデアルの交わりを Risa/Asir と Singular により計算するには以下のようにする.

Risa/Asir

```
[127] F=[t*(x-1),t*(y-1),(1-t)*(x-1),(1-t)*(y+1)];
[t*x-t,t*y-t,(-t+1)*x+t-1,(-t+1)*y-t+1]
[128] V=[t,x,y];
[t,x,y]
[129] G=nd_gr(F,V,0,2);
[-y^2+1,-x+1,-y+2*t-1]
```

Risa/Asir では, 関数 `nd_gr()` によりグレブナ基底を計算できる. 計算に使用する単項式順序は第 4 引数で指定する. それぞれ, 0 : 全次数逆辞書式順序 (grevlex), 1 : 全次数辞書式順序 (grlex), 2 : 辞書式順序 (lex) を意味する. 変数順序は第 2 引数の降順となる. 今の場合は変数順序 $t \succ x \succ y$ の lex 順序に関するグレブナ基底を出力する. 従って, 最後の出力から t を含まない多項式を取り出したものが求める交わりの生成元である.

Singular

```
> ring r=0,(t,x,y),lp;
> ideal I=x-1,y-1;
> ideal J=x-1,y+1;
> ideal F=t*I,(1-t)*J;
> groebner(F);
_[1]=y2-1
_[2]=x-1
_[3]=2t-y-1
> eliminate(F,t);
_[1]=y2-1
_[2]=x-1
> intersect(I,J);
_[1]=x-1
_[2]=y2-1
_[3]=xy-x-y+1
> option(redSB);
> groebner(_);
_[1]=y2-1
_[2]=x-1
```

Singular ではまず基礎環を宣言する. 1 行目で $\mathbf{Q}[t, x, y]$, 項順序を `lp` (辞書式順序) で宣言している. グレブナ基底の計算には関数 `groebner()` もしくは `std()` を用いる. Risa/Asir

の時と同じくこの出力から t を含まない多項式を取り出せばよい. また, Singular には消去イデアルを計算する関数 `eliminate()` があり, それを用いることもできる. 今の場合は第2引数で t を消去するように指定している. さらに, 直接交わりを計算する関数 `intersect()` も実装されている. ただし, この出力は簡約されていないので, `option(redSB)` で簡約基底を返すように指定してからグレブナ基底を計算している. 記号 "_"は直前の出力を表す.

2 つのイデアルの交わりが計算できたので, これを (D_1 の場合は 3 回) 繰り返せば計画イデアルは計算できる. しかし, 複数のイデアルの交わりをまとめて計算する方法 [3, chap 6.2] もある. N 個のイデアル $I_i = \langle f_{i1}, \ldots, f_{i,k_i} \rangle \subset k[x]$ の交わりは

$$I = \langle t_i f_{i1}, \ldots, t_i f_{ik_i}\,(i = 1, \ldots, N),\, t_1 + \cdots + t_N - 1 \rangle \cap k[x]$$

である. D_1 を例にすると,

Risa/Asir

```
[130] F=[t1*(x-1),t1*(y-1),t2*(x-1),t2*(y+1),
t3*(x+1),t3*(y-1),t4*(x+1),t4*(y+1),t1+t2+t3+t4-1];
[t1*x-t1,t1*y-t1,t2*x-t2,t2*y+t2,t3*x+t3,t3*y-t3,t4*x+t4,t4*y+t4,
t1+t2+t3+t4-1]
[131] V=[t1,t2,t3,t4,x,y];
[t1,t2,t3,t4,x,y]
[132] G=nd_gr(F,V,0,[[0,4],[0,2]]);
[-y^2+1,-x^2+1,(-y+1)*x+y+4*t4-1,(y+1)*x-y+4*t3-1,
(y-1)*x+y+4*t2-1,(-y-1)*x-y+4*t1-1]
```

最後の出力の多項式のうち, $t_1, \ldots, t_4$ を含まないものが求める交わりの生成元である. ここで, 項順序の指定法が以前とは少し異なっている. (2 のままでも構わない.) `[[0,4],[0,2]]` は, まず変数リスト `V` の前から 4 つ目までの変数に関する 0 : grevlex 順序で比較を行い, 引き分けた場合は次の 2 つの変数に関する 0 : grevlex 順序で比較を行え, という意味である (block order). つまり, $t_1 \succ t_2 \succ t_3 \succ t_4 \gg x \succ y$ という変数順序が指定されており, 各ブロック内では全次数逆辞書式順序が用いられる.

Singular

```
> ring r=0,(t1,t2,t3,t4,x,y),(dp(4),dp(2));
> ideal I=x-1,y-1;
> ideal J=x-1,y+1;
> ideal K=x+1,y-1;
> ideal L=x+1,y+1;
> ideal F=t1*I,t2*J,t3*K,t4*L,t1+t2+t3+t4-1;
> groebner(F);
_[1]=y^2-1
_[2]=x^2-1
_[3]=4*t4-x*y+x+y-1
_[4]=2*t3-t4*x+t4+x-1
_[5]=2*t2-t3*y+t3-t4*y+t4+y-1
```

```
_[6]=t1+t2+t3+t4-1
> eliminate(F,t1*t2*t3*t4);
_[1]=y^2-1
_[2]=x^2-1
> intersect(I,J,K,L);
_[1]=y^2-1
_[2]=x^2-1
_[3]=x^2*y+x^2-y-1
> option(redSB);
> groebner(_);
_[1]=y^2-1
_[2]=x^2-1
```

同じ事を Singular で行うにはこのように入力する. block order の指定は `(dp(4),dp(2))` のように項順序 (`dp` は grevlex 順を表す) の後にカッコで各ブロックの変数の数を入力する. また, `eliminate()` で複数の変数を消去したいときは変数の積を第 2 引数にとる. 更に, `intersect()` は 3 つ以上のイデアルの交わりも計算できる.

問 2 次の実施計画の計画イデアル (のグレブナ基底) を計算せよ.

参考文献 : [1, chap 3.2]

1. 問 1. の D_1
2. 問 1. の $D_2 \sim D_6$
3. $D_7 = \{(0,0),(1,0),(2,0),(3,0),(0,1),(1,1),(2,1),(0,2)\} \subset \mathbf{Q}^2$ (Echelon design)

問 3 (計算機に詳しい人向け) Singular の `eliminate()` や `intersect()` に相当する関数を Risa/Asir で実装せよ.

問 4 (計算機に詳しい人向け) 計画 D (例えば点のリスト) を入力とし, その計画イデアル $\mathbf{I}(D)$ を返す関数を Risa/Asir, Singular 等で実装せよ.

注 1 CoCoA には 問 4 に相当する関数 `IdealOfPoints()` が実装されており, 次のようにして利用する.

CoCoA

```
Use R ::= Q[x,y];
P := [[1,1],[1,-1],[-1,1],[-1,-1]];
I := IdealOfPoints(P);
I;
Ideal(y^2 - 1, x^2 - 1)
-------------------------------
I.GBasis;
```

```
[y^2 - 1, x^2 - 1]
-------------------------------
```

2 別名関係とイデアル所属問題

計画 D における 2 つの効果 (主効果 or 交互作用) F,G が別名関係を持つことと $F-G \in \mathbf{I}(D)$ であることは同値である．ここでは計算機により, イデアル所属を判定してみる．それには, グレブナ基底の基本的な性質のひとつである次の命題を用いる．

命題 3 ([2] chap 2.6) G をイデアル $I \subset k[x]$ のグレブナ基底とし, $f \in k[x]$ とする．このとき, $f \in I$ であることと, f を G で割った余りが 0 であることは同値.

Risa/Asir と Singular で余り (normal form) を計算するには次のようにする．

例 3 $f = x^2y + xy^2 + y^2$, $g = x^2y + xy^2 + y^2 - x^2 - 2y$ がイデアル $I = \langle y^2 - 1, xy - 1 \rangle$ に属するかどうか.

Risa/Asir

```
[133] B=[y^2-1,x*y-1];
[y^2-1,y*x-1]
[134] V=[x,y];
[x,y]
[135] GB=nd_gr(B,V,0,0);
[x-y,y^2-1]
[136] F=x^2*y+x*y^2+y^2;
y*x^2+y^2*x+y^2
[137] p_nf(F,GB,V,0);
2*y+1
[138] G=y*x^2+y^2*x+y^2-x^2-2*y;
(y-1)*x^2+y^2*x+y^2-2*y
[139] p_nf(G,GB,V,0);
0
```

まずグレブナ基底を計算し, `p_nf()` によって余り (の定数倍) を計算する．`p_nf()` の第 4 引数は割り算に使用する項順序の指定であるのでグレブナ基底計算に用いた項順序と同じものを指定しないとイデアル所属が正しく判定されない．この計算によって, $f \notin I$, $g \in I$ であることが分かった. Singular では次のようにする．

Singular

```
> ring r=0,(x,y),dp;
> ideal I=y^2-1,x*y-1;
> ideal GB=groebner(I);
```

```
> poly f=x^2*y+x*y^2+y^2;
> reduce(f,GB);
2y+1
> poly g=y*x^2+y^2*x+y^2-x^2-2*y;
> reduce(g,GB);
0
```

問 5 D_4 において, 共変量行列を考える.

1. このとき, 主効果 C と交互作用 AD は同じ列を生成することを確かめよ.
2. 主効果 C と交互作用 AD が別名関係にあることをイデアル所属を判定することで確かめよ.
3. $\{A^{e_1}B^{e_2}C^{e_3}D^{e_4} \mid e_i = 0, 1\}$ の間の別名関係を全て列挙せよ.

問 6 D_6 について,

1. AB と AC, AB と DE, AC と $ABEF$ がそれぞれ別名関係にあるかどうか判定せよ.
2. 別名関係を全て列挙するプログラムを書け.

問 7 D_4 について, 主効果のみからなるモデル $M : A/B/C/D$ を考える.

1. 問 5 で列挙した別名関係に注意して, M に 2 因子交互作用を 1 つ追加した識別可能な階層モデルを全て挙げよ.
2. M に 3 因子交互作用を追加して識別可能な階層モデルを作ることができるか.
3. M に交互作用を追加して得られる識別可能な階層モデルのグラフを作れ. (1. の続きを飽和モデルになるまで行え.)

参考文献

[1] Pistone, G. Riccomagno, E. Wynn, H. P. (2000). *Algebraic Statistics, Computational Commutative Algebra in Statistics.* Chapman & Hall.

[2] D. Cox, J. Little and D. O'Shea. (1992). *Ideals,Varieties, and Algorithms*, Springer-Verlag, New York.

[3] T. Becker, V. Weispfenning. (1993). *Gröbner Bases*, Springer-Verlag, New York-Berlin-Heidelberg.

[4] Asir User's Manual,
`http://www.math.kobe-u.ac.jp/OpenXM/Current/doc/asir2000/html-jp/man_toc.html`

[5] Singular Online Manual,
`http://www.singular.uni-kl.de/Manual/latest/index.htm`