2 Solutions

In the following pages, computer input is shown inside a box, while computer output has been centered and typed in smaller font.

Problem 1 (Ideal intersection)

The intersection is computed in the following manner. The first step is defining $E = \langle s(y-1-3x), (1-s)(y-2+x)\rangle$. Then the ideal intersection consists in computing $E \cap \mathbb{R}[x, y]$.

```
Use T::=Q[s,x,y];
I:=Ideal(s*(y-1-3*x),(1-s)*(y-2+x));
E:=Elim(s,I);
E;
```

Problem 1: First CoCoAcode.

```
Ideal(3/4x^2 + 1/2xy - 1/4y^2 - 5/4x + 3/4y - 1/2)
-------------------------------
```

A further check consists in verifying that $\langle y-1-3x\rangle\langle y-2+x\rangle$ equals $E \cap \mathbb{R}[x, y]$.

```
J:=Ideal(y-1-3*x);
L:=Ideal(y-2+x);
J*L;
E=J*L;
```

Problem 1: Second CoCoAinput code.

```
Ideal(-3x^2 - 2xy + y^2 + 5x - 3y + 2)
-------------------------------
TRUE
-------------------------------
```

Problem 2 (Gaussian elimination)

Firstly, the ideal I of system (1) is defined, then Gröbner basis and normal forms are computed.

```
Use T::=Q[x,y,z,w];
I:=Ideal(x+2y-2z+w+1,x+y+z-w-2);
GBasis(I);
NF(x,I);
NF(y,I);
NF(z,I);
NF(w,I);
```

Problem 2: CoCoAcode.

The (reduced) Gröbner basis for I is already in triangular form, but still not in terms of the parameters only. These are read from the normal forms.

```
[x + 2y - 2z + w + 1, -y + 3z - 2w - 3]
-------------------------------
-4z + 3w + 5
-------------------------------
3z - 2w - 3
-------------------------------
z
-------------------------------
w
-------------------------------
```

Note that by playing with different term orders, different parametrizations can be obtained. Try inputting `Use T::=Q[x,y,z,w], Xel;` What do you obtain?

Problem 3 (Linear parametrization)

The parameterized surface defines the ideal I. Then the elimination ideal $I \cap \mathbb{R}[x, y, z]$ retrieves the implicit equation required.

```
Use T::=Q[x,y,z,u,v];
I:=Ideal(x-1-u+v,y-u-2v,z+1+u-v);
J:=Elim(v,Elim(u,I));
J;
```

Problem 3: CoCoAcode.

```
Ideal(x + z)
-------------------------------
```

Problem 4 (Nonlinear parametrization)

Converting the parametric system of the *Folium* into an implicit equation involves eliminating the parameter t. However, a trick is required, as each parametric equation in (3) is a fractional polynomial. Transforming fractional polynomials into polynomials is achieved by defining an extra variable $b = \frac{1}{1+t^3}$, which is written as a polynomial condition $b(t^3+1) - 1 = 0$. In this form, the parametric system is translated into an equivalent polynomial system. Call F the ideal defining the *Folium*, i.e.

$$F = \langle 3atb - x, 3at^2b - y, b(t^3+1) - 1 \rangle \subset \mathbb{R}[b, a, t, x, y].$$

```
Use T::=Q[b,a,t,x,y];
F:=Ideal(3*a*t*b-x,3*a*t^2*b-y,b*(t^3+1)-1);
F;
Elim(b,Elim(t,F));
```

Problem 4: CoCoAcode for the *Folium*.

What we require is eliminating the parameter t and the extra variable b from F, i.e. construct the elimination ideal $F^* = F \cap \mathbb{R}[a, x, y]$. This elimination ideal will define a generator system of polynomials in terms only of a, x, y.

```
Ideal(3bat - x, 3bat^2 - y, bt^3 + b - 1)
-------------------------------
Ideal(axy - 1/3x^3 - 1/3y^3)
-------------------------------
```

For the *Cissoid*, the procedure mimics what was done above, i.e. converting the parametric system (4) into a system of polynomial equations and then eliminating. A suitable extra variable b is used.

```
Use T::=Q[b,a,t,x,y];
F:=Ideal(2*a*t^2*b-x,2*a*t^3*b-y,b*(t^2+1)-1);
Elim(b,Elim(t,F));
```

Problem 4: CoCoAcode for the *Cissoid*.

```
Ideal(1/2x^3 - ay^2 + 1/2xy^2)
-------------------------------
```

Problem 5 (Nonlinear parametrization)

```
Use T::=Q[t,x,y];
F:=Ideal(x*(1+t)-t,(y-1)*t^2-1);
Elim(t,F);
```

Problem 5: CoCoAcode.

```
Ideal(x^2y - 2x^2 + 2x - 1)
-------------------------------
```

Problem 6 (Nonlinear solving)

First, the ideal I generated by the two polynomials is defined. Two important issues are worth noting here. Firstly, the system (5) has a finite number of solutions. This is verified as, for each variable, we have one leading term (of the ideal) that depends only on it (i.e. x^2, y^3, see below). Secondly, the number of solutions is counted by those monomials not divisible by x^2, xy, y^3. How many monomials are? This number of monomials/solutions is counted by the *Hilbert function* (try `Hilbert(T/I);`).

```
Use T::=Q[x,y];
I:=Ideal(x^2+2y^2-3,x^2+xy+y^2-3);
I;
LT(I);
```

Problem 6: CoCoAcode for definition of I.

```
Ideal(x^2 + 2y^2 - 3, x^2 + xy + y^2 - 3)
------------------------------
Ideal(x^2, xy, y^3)
-------------------------------
```

The elimination ideals $J = I \cap \mathbb{R}[x]$ and $M = I \cap \mathbb{R}[y]$ are then constructed. Then the generators of J and M are factorized.

```
J:=Elim(y,I);
M:=Elim(x,I);
J:=Gens(J);
M:=Gens(M);
Factor(J[1]);
Factor(M[1]);
```

Problem 6: CoCoAcode for elimination ideal.

```
[[x^2 - 3, 1], [x - 1, 1], [x + 1, 1], [-1/2, 1]]
-------------------------------
[[y, 1], [y - 1, 1], [y + 1, 1], [3, 1]]
-------------------------------
```

In other words, the solution of (5) is a subset of $W = \{-\sqrt{3}, -1, 1, \sqrt{3}\} \otimes \{-1, 0, 1\}$. The last task is to evaluate which elements of the candidate set C are solutions of Equation (5). This is achieved by defining a new ring in which a plays the role of $\sqrt{3}$.

```
Use T::=Q[x,y,a];
I:=Ideal(a*a-3); -- a is sqrt(3)
L:=[-a,-1,1,a];
M:=[-1,0,1];
W:=L><M; -- candidate set of solutions
P1:=x^2+2y^2-3;
P2:=x^2+xy+y^2-3;
Foreach J In W Do
PrintLn(J," ",NF(Eval(P1,J),I)," ",NF(Eval(P2,J),I));
EndForeach;
```

Problem 6: CoCoAcode for evaluating the candidate solutions.

The solutions are $\pm(\sqrt{3}, 0), \pm(1, 1)$. Why `NF` was used to evaluate the candidate solutions?

Problem 7 (Nonlinear optimization)

The solution is based in elimination theory and Lagrange multipliers. First create the Lagrangian $L = F + mS$, where m is the Lagrange multiplier, F is the function to optimize and S is the restriction. Then the gradient ideal is created $I = \langle \nabla L \rangle \subset \mathbb{R}[m, x, y, z]$. Recall that ∇ refers to partial derivatives with respect to x, y, z and m. Then candidates for the extreme points are obtained by factorizing the generators of elimination ideals $E_1 = I \cap \mathbb{R}[x]$, $E_2 = I \cap \mathbb{R}[y]$ and $E_3 = I \cap \mathbb{R}[z]$.

```
Use T::=Q[m,x,y,z], Lex;
S:=x^2+y^2+z^2-4;
F:=x^3+y^3+z^3;
L:=F+m*S;
W:=[];
Foreach I In Indets() Do W:=[W,Der(L,I)]; W:=Flatten(W);
EndForeach;
I:=Ideal(W); J:=Elim(m,I);
E1:=Elim(y,Elim(z,J));
E2:=Elim(x,Elim(z,J));
E3:=Elim(x,Elim(y,J));
F1:=Gens(E1);
F2:=Gens(E2);
F3:=Gens(E3);
Factor(F1[1]);
Factor(F2[1]);
Factor(F3[1]);
```

Problem 7: CoCoAcode.

```
[[x, 1], [3x^2 - 4, 1], [x - 2, 1], [x + 2, 1], [x^2 - 2, 1], [-1/8, 1]]
-------------------------------
[[y, 1], [3y^2 - 4, 1], [y - 2, 1], [y + 2, 1], [y^2 - 2, 1], [1/8, 1]]
-------------------------------
[[z, 1], [3z^2 - 4, 1], [z - 2, 1], [z + 2, 1], [z^2 - 2, 1], [1/8, 1]]
-------------------------------
```

The candidate set is $\left\{0, \pm\frac{2}{\sqrt{3}}, \pm\sqrt{2}, \pm 2\right\}^3$, which has 343 candidates. However by using `Hilbert(T/I);` we find that there are only 14 solutions.

Problem 8 (Nonlinear optimization)

The only change with respect to Problem 7 is the inclusion of another polynomial restriction and thus for instance $E_1 = I \cap \mathbb{R}[x]$ is constructed by eliminating m, n, y and z.

```
Use T::=Q[n,m,x,y,z], Lex;
S:=x^2+y^2+z^2-4;
H:=x+y+z-1;
F:=x^3+y^3+z^3;
L:=F+m*S+n*H;
W:=[];
Foreach I In Indets() Do W:=[W,Der(L,I)]; W:=Flatten(W);
EndForeach;
I:=Ideal(W);
J:=Elim(n,Elim(m,I));
E1:=Elim(y,Elim(z,J));
E2:=Elim(x,Elim(z,J));
E3:=Elim(x,Elim(y,J));
E1; E2; E3;
```

Problem 8: CoCoAcode, compare with Problem 7.

```
Ideal(3/4x^4 - x^3 - 43/24x^2 + 17/12x + 7/8)
-------------------------------
Ideal(3/4y^4 - y^3 - 43/24y^2 + 17/12y + 7/8)
-------------------------------
Ideal(-9/11z^4 + 12/11z^3 + 43/22z^2 - 17/11z - 21/22)
-------------------------------
```

The candidate set this time is $\{\frac{1}{3} \pm \frac{\sqrt{22}}{6}, \frac{1}{3} \pm \frac{\sqrt{22}}{3}\}^3$. Of the 64 candidate points, only 6 are solutions.

Problem 9 (Nonlinear optimization)

```
Use T::=Q[r,x,y];
F:=x^2*y; G:=x^2+y^2-3;
L:=F-r*G;
I:=Ideal(Der(L,x),Der(L,y),G);
Hilbert(T/I);
E1:=Gens(Elim(x,Elim(r,I)));
E2:=Gens(Elim(y,Elim(r,I)));
Factor(E1[1]);
Factor(E2[1]);
```

Problem 9: CoCoAcode.

```
WARNING! HilbertPoincare input not homogeneous: computing LT...
H(0) = 1
H(1) = 3
H(2) = 2
H(t) = 0   for t >= 3
-------------------------------
[[y^2 - 3, 1], [y - 1, 1], [y + 1, 1], [-1/2, 1]]
-------------------------------
[[x, 1], [x^2 - 2, 1]]
-------------------------------
```

Out of the 12 elements of the candidate set $\{0, \pm\sqrt{2}\} \otimes \{\pm 1, \pm\sqrt{3}\}$, only 6 are solutions (counted by `Hilbert(T/I);`).

Problem 10 (Nonlinear optimization)

Firstly, `Maple` code is used to construct the Lagrangian $L = f - rg$, where r is the Lagrange multiplier and $f, g \in \mathbb{R}[x_1, x_2]$. After differentiation of L, the expressions to be simultaneously solved are $\frac{n_1}{d_1} = 0, \frac{n_2}{d_2} = 0, g = 0$.

```
restart:  f:=int(1/(t^4+1),t=x[1]..x[2]):  g:=x[1]^2*x[2]^2-1:
L:=f-r*g:  ## Lagrangian
p1:=diff(L,x[1]):  p2:=diff(L,x[2]):
n1:=numer(factor(p1));
n2:=numer(factor(p2));
d1:=expand(denom(factor(p1)));
d2:=expand(denom(factor(p2)));
```

Problem 10: First `Maple` code.

$$n1 := -1 - 2\,r\,x_1\,{x_2}^2 - 2\,r\,{x_1}^5\,{x_2}^2$$

$$n2 := 1 - 2\,r\,{x_1}^2\,{x_2}^5 - 2\,r\,{x_1}^2\,x_2$$

$$d1 := 1 + {x_1}^4$$

$$d2 := {x_2}^4 + 1$$

Saturation with respect to $d_1 d_2$ is carried out adding the polynomial $d_1 d_2 v + 1$ together with numerators n_1, n_2. Then a Gröbner basis is computed using the `Maple`package `Groebner`. It is important to select a lexical term ordering in which x_1 and x_2 are the smallest underterminates.

```
with(Groebner):
B:=Basis([n1,n2,d1*d2*v+1],plex(v,r,x[1],x[2]));
solve({B[1],g},{x[1],x[2]}):  evalf(%);
```

Problem 10: Second Maplecode.

$$B := [{x_2}^4\,x_1 + x_2\,{x_1}^4 + x_2 + x_1,\ -1 - 2\,{x_2}^4 + 2\,r\,{x_2}^3 - x_2\,{x_1}^3 - {x_2}^8 + {x_1}^2\,{x_2}^2 \\ +2\,r\,{x_2}^7 - {x_2}^5\,{x_1}^3, -{x_2}^3\,{x_1}^3 - {x_2}^2 + 2\,r\,x_1 + 2\,r\,x_2 + 2\,r\,{x_2}^4\,x_1 - {x_2}^6 + {x_1}^2 + 2\,{x_2}^5\,r, \\ -2\,r\,{x_2}^4 + {x_2}^5 + {x_1}^2\,{x_2}^3 + {x_2}^2\,{x_1}^3 + 2\,r\,{x_1}^4 + {x_1}^5 + x_2 + x_1,\ -4\,r^2\,{x_2}^3\,{x_1}^3 + v]$$

$$\{x_1 = 1.,\, x_2 = -1.\},\, \{x_1 = -1.,\, x_2 = 1.\},\, \{x_1 = 1.\,I,\, x_2 = -1.\,I\}, \\ \{x_2 = 0.7071067812 - 0.7071067812\,I,\, x_1 = 0.7071067812 + 0.7071067812\,I\}, \\ \{x_2 = -0.7071067812 + 0.7071067812\,I,\, x_1 = 0.7071067812 + 0.7071067812\,I\}$$

Problem 11 (Design of Experiments)

The Latin Square in (7) is associated with the following design model matrix

cell	r_1	r_2	r_3	c_1	c_2	c_3	t_1	t_2	t_3
11	1	0	0	1	0	0	1	0	0
12	1	0	0	0	1	0	0	1	0
13	1	0	0	0	0	1	0	0	1
21	0	1	0	1	0	0	0	0	1
22	0	1	0	0	1	0	1	0	0
23	0	1	0	0	0	1	0	1	0
31	0	0	1	1	0	0	0	1	0
32	0	0	1	0	1	0	0	0	1
33	0	0	1	0	0	1	1	0	0

where $r_1, \ldots, r_3$ indicate rows, $c_1, \ldots, c_3$ indicate columns and $t_1 = A, t_2 = B$ and $t_3 = C$ are the treatments.

```
Use T::=Q[r[1..3],c[1..3],t[1..3]], DegLex;
D:=[[1,0,0,1,0,0,1,0,0],
[1,0,0,0,1,0,0,1,0],
[1,0,0,0,0,1,0,0,1],
[0,1,0,1,0,0,0,0,1],
[0,1,0,0,1,0,1,0,0],
[0,1,0,0,0,1,0,1,0],
[0,0,1,1,0,0,0,1,0],
[0,0,1,0,1,0,0,0,1],
[0,0,1,0,0,1,1,0,0]];
I:=IdealOfPoints(D);
QuotientBasis(I);
```

Problem 11: CoCoAcode.

In this example is specially important to use a *graded* term ordering in order to include as many linear factors as possible. Does the result make sense with the structure of the LS?

```
-------------------------------
[1, t[3], t[2], c[3], c[3]t[3], c[3]t[2], c[2], r[3], r[2]]
-------------------------------
```

What happens if in the first line of the analysis you type instead
`Use T::=Q[r[1..3],c[1..3],t[1..3]],Lex;`
or `Use T::=Q[r[1..3],c[1..3],t[1..3]], Xel;`?

Problem 12 (Design of Experiments)

The design ideal $I(\mathcal{D})$ is the set of polynomials that vanish at the design points. The Hilbert function describes the number of monomials for each degree in the model.

```
Use T::=Q[x[1..2]];
D:=[[1,1],[1,-1],[-1,1],[-1,-1]];
I:=IdealOfPoints(D);
QuotientBasis(I);
Hilbert(T/I);
```

Problem 12: CoCoAcode.

```
[1, x[2], x[1], x[1]x[2]]
-------------------------------
WARNING! HilbertPoincare input not homogeneous: computing LT...
H(0) = 1
H(1) = 2
H(2) = 1
H(t) = 0   for t >= 3
-------------------------------
```

Problem 13 (Design of Experiments)

We first verify two ways of constructing $I(D)$: using the point coordinates (ideal I) and using the grid equations (ideal J). Then its equality is verified. What type of structure can you see in the monomials for the model?

```
Use T::=Q[x[1..3]];
D:=[[1,1,1],[1,-1,1],[-1,1,1],[-1,-1,1],
[1,1,-1],[1,-1,-1],[-1,1,-1],[-1,-1,-1]];
I:=IdealOfPoints(D);
J:=Ideal(x[1]^2-1,x[2]^2-1,x[3]^2-1);
I=J;
QuotientBasis(I);
```

Problem 13: First CoCoAcode.

```
TRUE
-------------------------------
[1, x[3], x[2], x[2]x[3], x[1], x[1]x[3], x[1]x[2], x[1]x[2]x[3]]
-------------------------------
```

Now we solve the problem. The ideal F is constructed by adding the generator, while G is constructed by the semicolon operator.

```
F:=Ideal(x[1]^2-1,x[2]^2-1,x[3]^2-1,x[1]*x[2]*x[3]-1);
G:=J:Ideal(x[1]*x[2]*x[3]+1);
G=F;
```

Problem 13: Second CoCoAcode.

```
TRUE
-------------------------------
[1, x[3], x[2], x[1]]
-------------------------------
```

What changes in the above results if you redo computations with a different term order, for instance `Use T::=Q[x[1..3]], Lex;` or `Use T::=Q[x[1..3]], Xel;`?

Problem 14 (Design of Experiments)

```
Use T::=Q[x[1..2]], Ord(Mat([[1,1],[0,1]]));
D1:=[[3,1],[2,3],[0,2],[1,0]];
I:=IdealOfPoints(D1);
QuotientBasis(I);
```

Problem 14a: CoCoAcode.

```
Use T::=Q[x[1..2]], Ord(Mat([[1,1],[0,1]]));
Use T::=Q[x[1..2]], Ord(Mat([[1,1],[1,0]]));
Use T::=Q[x[1..2]], Ord(Mat([[1,0],[0,1]]));
```

Problem 14: Different matrix term orders in CoCoAcode.

A term order in $\mathbb{R}[x_1, \ldots, x_d]$ can be associated with a square matrix of integer entries and of size d, that satisfies the following conditions: a)the first entry in each column is a positive number and b)the matrix is full rank. Defining a term order in this form is easily achieved in CoCoA, see code above. Below are part of the solutions for Problem 14a.

```
[1, x[2], x[1], x[1]^2]
-------------------------------
[1, x[2], x[2]^2, x[1]]
-------------------------------
[1, x[2], x[2]^2, x[2]^3]
-------------------------------
```

Try using other matrices and redo computations. Can you obtain other models?

```
Use T::=Q[x[1..2]], Ord(Mat([[0,1],[1,0]]));
D2:=[[0,0],[1,0],[0,1],[-1,1]];
I:=IdealOfPoints(D2);
QuotientBasis(I);
```

Problem 14b: CoCoAcode.

```
Use T::=Q[x[1..2]];
D3:=[[1,1],[1,-1],[-1,1],[-1,-1],[0,0]];
I:=IdealOfPoints(D3);
QuotientBasis(I);
```

Problem 14c: CoCoAcode.

Now for every design, plot the model exponents. What do you conclude?

Problem 15 (Maximum likelihood)

```
Use T::=Q[t[1..2]];
P1:=t[1];
P2:=t[2];
P3:=(t[1]-1)^2+(t[2]-1)^2;
P4:=(t[1]+1)^2+2*(t[2]-2)^2-9;
L:=[P1,P2,P3,P4];
M1:=[]; M2:=[];
Foreach I In L Do M1:=[M1,Der(I,t[1])]; M2:=[M2,Der(I,t[2])];
EndForeach;
M1:=Flatten(M1);
M2:=Flatten(M2);
```

Problem 15: CoCoAcode with definitions: $\theta_1 =$`t[1]`, $\theta_2 =$`t[2]`.

Maximum likelihood estimates for θ_1, θ_2 are obtained by the simultaneous solution of the log-likelihood $l(\theta_1, \theta_2)$. The log-likelihood is defined as the logarithm of the joint probability (taking out the multinomial coefficient that does not depend on the parameters):

$$l(\theta_1, \theta_2) = \ln\left(\prod_{i=1}^{4} p_i^{x_i}\right) = \sum_{i=1}^{4} x_i \ln(p_i)$$

By the chain rule,

$$\frac{\partial l(\theta_1, \theta_2)}{\partial \theta_j} = \sum_{i=1}^{4} x_i \frac{1}{p_i} \frac{\partial}{\partial \theta_j} p_i = \sum_{i=1}^{4} x_i \frac{1}{p_i} p_i^{(j)}, \tag{15}$$

where $p_i^{(j)} = \frac{\partial}{\partial \theta_j} p_i$ and $j = 1, 2$. The MLEs are the simultaneous solutions in (θ_1, θ_1) of the system created by equating (15) to zero for $j = 1, 2$, that is

$$\begin{aligned} \frac{x_1}{p_1} p_1^{(1)} + \frac{x_2}{p_2} p_2^{(1)} + \frac{x_3}{p_3} p_3^{(1)} + \frac{x_4}{p_4} p_4^{(1)} &= 0 \\ \frac{x_1}{p_1} p_1^{(2)} + \frac{x_2}{p_2} p_2^{(2)} + \frac{x_3}{p_3} p_3^{(2)} + \frac{x_4}{p_4} p_4^{(2)} &= 0 \end{aligned} \tag{16}$$

The above is a system of two fractional polynomial equations and as such, it is not possible to solve it directly with algebraic techniques. For instance, if we simultaneously solve the numerators, those solutions which make the

denominator zero must be avoided. In order to define Eq. (16), we note that the system can be written as

$$\begin{aligned}
\frac{1}{p_1p_2p_3p_4}\sum_{i=1}^{4} x_i p_i^{(1)} \frac{p_1p_2p_3p_4}{p_i} &= 0 \\
\frac{1}{p_1p_2p_3p_4}\sum_{i=1}^{4} x_i p_i^{(2)} \frac{p_1p_2p_3p_4}{p_i} &= 0
\end{aligned}$$

```
X:=[2,1,9,1]; -- Data
N1:=0; N2:=0; A:=Product(L);
Foreach I In 1..4 Do N1:=N1+A*X[I]*M1[I]/L[I];
N2:=N2+A*X[I]*M2[I]/L[I];
EndForeach;
J:=Ideal(N1,N2); J;
```

Problem 15: CoCoAcode for defining J.

```
Ideal(22t[1]^4t[2] + 44t[1]^2t[2]^3 + 4t[2]^5 + 16t[1]^3t[2]
 - 168t[1]^2t[2]^2 - 38t[1]t[2]^3 - 24t[2]^4 - 40t[1]^2t[2]
 + 164t[1]t[2]^2 + 40t[2]^3 + 12t[1]t[2] - 32t[2]^2, t[1]^5
 + 25t[1]^3t[2]^2 + 42t[1]t[2]^4 - 36t[1]^3t[2] + 26t[1]^2t[2]^2
 - 208t[1]t[2]^3 - 2t[1]^3 - 8t[1]^2t[2] + 188t[1]t[2]^2 + 4t[1]^2
 - 32t[1]t[2])
```

We remove the zero set of the numerator through ideal saturation. First, define the ideal of the numerators $J = \langle n_1, n_2\rangle$, then saturate it by the common denominator $p_1p_2p_3p_4$, i.e. $B = J : (p_1p_2p_3p_4)^\infty$. The final step involves finding the solution set of B.

```
B:=Saturation(J,Ideal(A)); B;
E1:=Gens(Elim(t[2],B));
E2:=Gens(Elim(t[1],B));
E1; E2;
```

Problem 15: CoCoAcode for saturating J and isolating its univariate generators.

```
Ideal(t[2]^4 - 3/46t[1]^3 + 24/23t[1]^2t[2] + 113/46t[1]t[2]^2
- 116/23t[2]^3 + 1/23t[1]^2 - 80/23t[1]t[2] + 98/23t[2]^2
+ 3/23t[1] - 8/23t[2], t[1]^4 + 13/23t[1]^3 - 36/23t[1]^2t[2]
+ 100/23t[1]t[2]^2 - 12/23t[2]^3 - 38/23t[1]^2 - 74/23t[1]t[2]
+ 8/23t[2]^2 + 16/23t[1], t[1]^2t[2]^2 + 2/23t[1]^3 - 72/23t[1]^2t[2]
- 75/23t[1]t[2]^2 + 4/23t[2]^3 - 2/23t[1]^2 + 130/23t[1]t[2]
+ 8/23t[2]^2 - 2/23t[1] - 16/23t[2], t[1]t[2]^3 + 47/23t[1]^3
+ 25/23t[1]^2t[2] - 70/23t[1]t[2]^2 - 29/23t[2]^3 + 26/23t[1]^2
- 2t[1]t[2] + 136/23t[2]^2 - 80/23t[1] - 118/23t[2] + 32/23,
t[1]^3t[2] - 95/23t[1]^3 - 10/23t[1]^2t[2] - 42/23t[1]t[2]^2
+ 54/23t[2]^3 - 54/23t[1]^2 + 102/23t[1]t[2] - 256/23t[2]^2
+ 160/23t[1] + 236/23t[2] - 64/23)
[651358152528090410753413/1299678801495263028496Ot[1]^10
- 2860311887188570934178031/12996788014952630284960t[1]^9
- 4630922517312189101783717/12996788014952630284960t[1]^8
+ 5070009755844296008988321 1/12996788014952630284960t[1]^7
- 2532981378113684286173441/6498394007476315142 48t[1]^6
- 5279610911652774037281059/812299250934539392810t[1]^5
+ 5507734416832404916877403/406149625467269696405t[1]^4
- 14512589040399695480494707/1624598501869078785620t[1]^3
+ 10663158512150246847 08574/406149625467269696405t[1]^2
- 1352708516347586747879 88/406149625467269696405t[1]
+ 119173164080453647001 6/81229925093453939281]
[-31189231893847139912348 3339/49472875640041253397668 24t[2]^10
+ 47868690689252349517647 66029/49472875640041253397668 24t[2]^9
- 62100767602023164457991260139/98945751280082506795336 48t[2]^8
+ 209295024040268539842832712601/98945751280082506795336 48t[2]^7
- 185849146641774214236081838765/49472875640041253397668 24t[2]^6
+ 164193315669976352990535476955/49472875640041253397668 24t[2]^5
- 33221255919455309362996846595/24736437820020626698834 12t[2]^4
+ 50044350288644187199763466 61/24736437820020626698834 12t[2]^3
- 8793562084157237473938312 1/12368218910010313349417 06t[2]^2
- 17375585608437538146072 61/12368218910010313349417 06t[2]
+ 44581361407152118399160/6184109455005156674708 53]
```

What different information do `Hilbert(T/J);` and `Hilbert(T/B);` give? What would you do next?

Problem 15: code in Singular

```
ring r=0,(t,u),dp;
poly p1=t;
poly p2=u;
poly p3=(t-1)^2+(u-1)^2-2;
poly p4=(t+1)^2+2*(u-2)^2-9;
poly p11=diff(p1,t); poly p21=diff(p2,t);
poly p31=diff(p3,t); poly p41=diff(p4,t);
poly p12=diff(p1,u); poly p22=diff(p2,u);
poly p32=diff(p3,u); poly p42=diff(p4,u);
```

Problem 15: Singularcode with definitions: $\theta_1 =$t, $\theta_2 =$u.

```
int x1=2; int x2=1; int x3=9; int x4=1; // data
poly n1=x1*p11*p2*p3*p4+x2*p21*p1*p3*p4+x3*p31*p1*p2*p4
+x4*p41*p1*p2*p3;
poly n2=x1*p12*p2*p3*p4+x2*p22*p1*p3*p4+x3*p32*p1*p2*p4
+x4*p42*p1*p2*p3;
poly d=p1*p2*p3*p4;
ideal I=n1,n2;
I;
```

Problem 15: Singularcode for defining $I = \langle n_1, n_2 \rangle$.

```
I[1]=22t4u+44t2u3+4u5+16t3u-168t2u2-38tu3-24u4-48t2u+164tu2+32u3
I[2]=t5+25t3u2+42tu4-36t3u+26t2u2-208tu3-4t3-8t2u+176tu2
```

```
LIB "elim.lib";
ideal K=sat(I,d)[1];
LIB "solve.lib";
solve(K);
```

Problem 15: Singularcode for $K = I : (p_1p_2p_3p_4)^\infty$ and solving I.

```
K[1]=23t3-138t2u+69tu2-184u3+259t2-346tu+779u2+98t-152u-520
K[2]=1058u4+690t2u+2806tu2-5888u3+731t2-4718tu+6753u2+294t-456u-1560
K[3]=529tu3+3358t2u-4324tu2+2691u3-5986t2+10512tu-10232u2-3548t-3528u+12480
K[4]=529t2u2-1380t2u-1863tu2+460u3-518t2+3682tu-1558u2-196t+304u+1040
[1]:
   [1]:
      1.13777237
   [2]:
      -0.051178458
```

```
[2]:
   [1]:
      -1.786224
   [2]:
      -0.024407853
[3]:
   [1]:
      1.55266251
   [2]:
      0.67511145
[4]:
   [1]:
      -0.084833799
   [2]:
      1.06887818
[5]:
   [1]:
      1.07102236
   [2]:
      1.19387508
[6]:
   [1]:
      -3.69341014
   [2]:
      2.50117045
[7]:
   [1]:
      0.38671104
   [2]:
      3.76738267
[8]:
   [1]:
      (3.403802-i*1.875051)
   [2]:
      (3.60849729+i*2.45685114)
[9]:
   [1]:
      (3.403802+i*1.875051)
   [2]:
      (3.60849729-i*2.45685114)
```

There are 9 outputs, of which 2 are complex; and of the remaining seven real, three give positive values for θ_1 and θ_2. What would you do next?

Problem 16 (Contingency table)

The likelihood of the table is the multinomial likelihood Lik $= \prod_{i,j} p_{i,j}^{u_{i,j}}$, where $u_{i,j}$ are the observed counts for every cell in the table, and $p_{i,j}$ are the probabilities of cells in the table. To solve 16a, the probabilities are considered as parameters, while for 16b, the model (9) is used to parametrize the probabilities. In both cases the log-likelihood is maximised using Lagrange multipliers to restrict the search over the probability simplex.

```
R:=2; C:=3;
Use T::=Q[m,t[0..(R-1),0..(C-1)]];
U:=Mat([[12,3,7],[3,7,12]]); -- Data
P:=U*0;
P[1,1]:=t[0,0]; P[1,2]:=t[0,1]; P[1,3]:=t[0,2];
P[2,1]:=t[1,0]; P[2,2]:=t[1,1]; P[2,3]:=t[1,2];
S:=P[1,1]*P[1,2]*P[1,3]*P[2,1]*P[2,2]*P[2,3];
```

Problem 16: First CoCoAcode (Definitions for 16a).

Once again, Lagrange multipliers are used. Note that the Lagrangian is $L = \log(\text{Lik}) + m(\sum_{i,j} p_{i,j} - 1)$, where m is the Lagrange multiplier. The Lagrangian can be written as

$$L = \sum_{i,j} (u_{i,j} \log(p_{i,j}) + m p_{i,j}) - m.$$

The derivative L' with respect to a model parameter can be shown to be

$$L' = \frac{1}{S} \sum_{i,j} \left[(u_{i,j} + m p_{i,j}) \frac{S p'_{i,j}}{p_{i,j}} \right], \tag{17}$$

where $S = \prod_{i,j} p_{i,j}$, and the expression between square brackets in the sum is not a fractional polynomial.

Two double loops are required. The inner double loop computes the sum in Equation (17), while the outer double loop varies among all parameters, effectively computing $F = \nabla L$.

```
F:=[]; A:=-1;
Foreach I In 0..(R-1) Do
Foreach J In 0..(C-1) Do
A:=A+P[I+1,J+1]; W:=0;
Foreach I1 In 0..(R-1) Do
Foreach J1 In 0..(C-1) Do
W:=W+(U[I1+1,J1+1]
+m*P[I1+1,J1+1])*Der(P[I1+1,J1+1],t[I,J])*S/P[I1+1,J1+1];
EndForeach;
EndForeach;
F:=[F, W];
F:=Flatten(F);
EndForeach;
EndForeach;
F:=Flatten([F,A]);
```

Problem 16: Computing ∇L with CoCoA.

The final step is the gradient ideal $K = \langle \nabla L \rangle$, and the saturation by S to remove singularities and obtain directly ML estimates of the table probabilities.

```
K:=Ideal(F);
K:=Saturation(K,Ideal(S));
K;
```

Problem 16: Third CoCoAcode.

```
Ideal(t[1,2] - 3/11, t[1,1] - 7/44, t[1,0] - 3/44, t[0,2] - 7/44,
t[0,1] - 3/44, t[0,0] - 3/11, m + 44)
-------------------------------
```

Note that the MLEs obtained coincide with those by the method of moments. The log-linear model of Equation (9) corresponds to the following table:

$e^{\theta_{0,0}}$	$e^{\theta_{0,0}+\theta_{0,1}}$	$e^{\theta_{0,0}+\theta_{0,1}+\theta_{0,2}}$
$e^{\theta_{0,0}+\theta_{1,0}}$	$e^{\theta_{0,0}+\theta_{1,0}+\theta_{0,1}+\theta_{1,1}}$	$e^{\theta_{0,0}+\theta_{1,0}+\theta_{0,1}+\theta_{1,1}+\theta_{0,2}+\theta_{1,2}}$

This is specified by letting $t_{i,j} = e^{\theta_{i,j}}$ in the Definitions section of the code. The rest of the computations follow, with the only change of using a lower degree version of S for saturation.

```
P[1,1]:=t[0,0];
P[1,2]:=t[0,0]*t[0,1];
P[1,3]:=t[0,0]*t[0,1]*t[0,2];
P[2,1]:=t[0,0]*t[1,0];
P[2,2]:=t[0,0]*t[1,0]*t[0,1]*t[1,1];
P[2,3]:=t[0,0]*t[1,0]*t[0,1]*t[1,1]*t[1,2]*t[0,2];
S:=P[2,3];
```

Problem 16: code for 16b, to replace final part of the code in page 30.

```
Ideal(t[1,2] - 36/49, t[1,1] - 28/3, t[1,0] - 1/4, t[0,2] - 7/3,
t[0,1] - 1/4, t[0,0] - 3/11, m + 44)
-------------------------------
```

The MLE for the parameters are $\theta_{i,j} = \log(t_{i,j})$. How do you check this results?

Problem 17 (Conditional independence)

We use model

$$\log(p) = \theta_0 + \theta_1 x_1 + \theta_1 x_2 + \theta_3 x_3 + \theta_{13} x_1 x_3 + \theta_{23} x_2 x_3 \tag{18}$$

Note that after exponentiating we have

$$\begin{aligned} p &= \exp\left(\theta_0 + \theta_1 x_1 + \theta_1 x_2 + \theta_3 x_3 + \theta_{13} x_1 x_3 + \theta_{23} x_2 x_3\right) \\ &= \psi_0 \psi_1^{x_1} \psi_2^{x_2} \psi_3^{x_3} \psi_{13}^{x_1 x_3} \psi_{23}^{x_2 x_3} \end{aligned} \tag{19}$$

with $\psi_i = e^{\theta_i}$. Why Equation (19) implies $X_1 \perp\!\!\!\perp X_2 | X_3$?

Let X be the design-model matrix for Equation (18), after transforming the levels from $0, 1$ to ± 1:

	1	x_1	x_2	x_3	x_1x_3	x_2x_3
p_{111}	1	1	1	−1	−1	−1
p_{110}	1	1	1	−1	−1	−1
p_{101}	1	1	−1	1	1	−1
p_{100}	1	1	−1	1	1	−1
p_{011}	1	−1	1	−1	1	−1
p_{010}	1	−1	1	−1	1	−1
p_{001}	1	−1	−1	1	−1	−1
p_{000}	1	−1	−1	1	−1	−1

The rows of X are indexed by $p_{ijk} = p(X_1 = i, X_2 = j, X_3 = k)$. The next operation is to compute the kernel of X^T, using `Maple` commands.

```
restart:  with(LinearAlgebra):
X:=Matrix([[1,1,1,1,1,1],[1,1,1,-1,-1,-1],[1,1,-1,1,1,-1],
[1,1,-1,-1,-1,1],[1,-1,1,1,-1,1],[1,-1,1,-1,1,-1],
[1,-1,-1,1,-1,-1],[1,-1,-1,-1,1,1]]):
B:=NullSpace(Transpose(X));
```

Problem 17: First `Maple` code.

$$B := \left\{ \begin{bmatrix} 1 \\ 0 \\ -1 \\ 0 \\ -1 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ -1 \\ 0 \\ -1 \\ 0 \\ 1 \end{bmatrix} \right\}$$

But the rows of the kernel B are also indexed by p_{ijk}, giving the usual formulæ

$$p_{111}p_{001} - p_{101}p_{011} = 0 \tag{20}$$

$$p_{110}p_{000} - p_{100}p_{010} = 0 \tag{21}$$

That is, the ideal representing the independence property is generated by two quadratic polynomials: $I_{X_1 \perp\!\!\!\perp X_2|X_3} = \langle p_{111}p_{001} - p_{101}p_{011}, p_{110}p_{000} - p_{100}p_{010} \rangle$

We can saturate the independence ideal $I_{X_1 \perp\!\!\!\perp X_2|X_3} : \left(\prod_{i,j,k=0,1} p_{ijk}\right)^\infty$, by adding $v \prod_{i,j,k=0,1} p_{ijk} + 1$ to Equations (20) and (21) and eliminating v. What are the implications of the result?

```
r1:=p[1,1,0]*p[0,0,0]-p[1,0,0]*p[0,1,0]:
r2:=p[1,1,1]*p[0,0,1]-p[1,0,1]*p[0,1,1]:
r3:=p[1,1,1]*p[1,1,0]*p[1,0,1]*p[1,0,0]*p[0,1,1]*p[0,1,0]
p[0,0,1]*p[0,0,0]:
with(Groebner):
B:=Basis([r1,r2,v*r3+1],plex(v,p[1,1,1],p[1,1,0],p[1,0,1],
p[1,0,0],p[0,1,1],p[0,1,0],p[0,0,1],p[0,0,0]));
```

Problem 17: Second `Maple` code.

$B := [p_{1,1,0}p_{0,0,0} - p_{1,0,0}p_{0,1,0}, p_{1,1,1}p_{0,0,1} - p_{1,0,1}p_{0,1,1}, vp_{0,1,1}{}^2p_{1,0,1}{}^2p_{0,1,0}{}^2p_{1,0,0}{}^2 + 1]$

Problem 18 (Moment aliasing)

Let $[m_\alpha]_{\alpha\in L}$ be the vector of moments; $X = [x^\alpha]_{x\in D,\alpha\in L}$ be the non singular design-model matrix for L; and let $[p]$ be the vector of probabilities. Clearly $[m_\alpha] = X^T[p]$, that is

$$\begin{bmatrix} m_{0,0} \\ m_{1,0} \\ m_{0,1} \\ m_{1,1} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_{0,0} \\ p_{1,0} \\ p_{0,1} \\ p_{1,1} \end{bmatrix}$$

This is a simply linear relation between probabilities and moments, which can be easily inverted. For instance, if we have ML estimators for the probabilities, we can obtain ML estimators for the moments.

```
Use T::=Q[m[0..1,0..1]];
XT:=Mat([[1,1,1,1],[0,1,0,1],[0,0,1,1],[0,0,0,1]]);
M:=Mat([[m[0,0]],[m[1,0]],[m[0,1]],[m[1,1]]]);
P:=Inverse(XT)*M;
P;
```

Problem 18: First CoCoAcode.

```
Mat([
  [m[0,0] - m[0,1] - m[1,0] + m[1,1]],
  [m[1,0] - m[1,1]],
  [m[0,1] - m[1,1]],
  [m[1,1]]
])
-------------------------------
```

We then add the independence condition to create the sought ideal $I = \langle p_{00}p_{11} - p_{10}p_{01}, p_{00} + p_{10} + p_{01} + p_{11} - 1\rangle \subset \mathbb{R}[m_{0,0}, m_{1,0}, m_{0,1}, m_{1,1}]$. Interpret the result.

```
P00:=P[1,1]; P10:=P[2,1]; P01:=P[3,1]; P11:=P[4,1];
I:=Ideal(P00*P11-P10*P01,P00+P01+P10+P11-1);
I;
```

Problem 18: Second CoCoAcode.

```
Ideal(-m[0,1]m[1,0] + m[0,0]m[1,1], m[0,0] - 1)
-------------------------------
```

Problem 19 (Moment aliasing)

The relationship between moments and probabilities is given by

$$\begin{bmatrix} m_{0,0} \\ m_{1,0} \\ m_{0,1} \\ m_{1,1} \\ m_{0,2} \\ m_{1,2} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & -1 & -1 & 0 \\ 0 & 1 & 1 & 0 & -1 & -1 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_6 \end{bmatrix}$$

which is used as building block for the moment confounding structure.

```
Use T::=Q[m[0..1,0..2]];
XT:=Mat([[1,1,1,1,1,1],[1,1,0,-1,-1,0],[0,1,1,0,-1,-1],
[0,1,0,0,1,0],[0,1,1,0,1,1],[0,1,0,0,-1,0]]);
M:=Mat([[m[0,0]],[m[1,0]],[m[0,1]],[m[1,1]],[m[0,2]],
[m[1,2]]]);
P:=Inverse(XT)*M;
P;
```

Problem 19: First CoCoAcode.

```
Mat([
  [1/2m[0,0] - 1/2m[0,2] + 1/2m[1,0] - 1/2m[1,2]],
  [1/2m[1,1] + 1/2m[1,2]],
  [1/2m[0,1] + 1/2m[0,2] - 1/2m[1,1] - 1/2m[1,2]],
  [1/2m[0,0] - 1/2m[0,2] - 1/2m[1,0] + 1/2m[1,2]],
  [1/2m[1,1] - 1/2m[1,2]],
  [-1/2m[0,1] + 1/2m[0,2] - 1/2m[1,1] + 1/2m[1,2]]
])
-------------------------------
```

We start by only imposing the usual restriction on probabilities $I = \langle p_1 + p_2 + p_3 + p_4 + p_5 + p_6 - 1\rangle$. How do you interpret your result?

```
P1:=P[1,1]; P2:=P[2,1]; P3:=P[3,1];
P4:=P[4,1]; P5:=P[5,1]; P6:=P[6,1];
I:=Ideal(P1+P2+P3+P4+P5+P6-1);
I;
```

Problem 19: Second CoCoAcode.

```
Ideal(m[0,0] - 1)
-------------------------------
```

The independence is coded in the usual form, i.e. $\Pr(X_1 = x_1, X_2 = x_2) = \Pr(X_1 = x_1)\Pr(X_2 = x_2)$. This is done for all probabilities, for example $p_1 = (p_1 + p_2)(p_1 + p_4)$. The condition $\sum_{i=1}^{6} p_i = 1$ is added to create the independence ideal I, whose Gröbner basis has 11 elements.

```
I:=Ideal(P1+P2+P3+P4+P5+P6-1,P1-(P1+P2)*(P1+P4),
P2-(P1+P2)*(P2+P3),P3-(P3+P6)*(P2+P3),P4-(P4+P5)*(P1+P4),
P5-(P4+P5)*(P5+P6),P6-(P3+P6)*(P5+P6));
GBasis(I);
```

Problem 19: Third CoCoAcode.

```
[m[0,0] - 1, -1/2m[0,1]m[0,2] - 1/2m[0,2]^2 + 1/2m[0,1]m[1,1]
+ 1/2m[0,2]m[1,1] + 1/2m[0,1] + 1/2m[0,2] - 1/2m[1,1] - 1/2m[1,2],
-2m[0,2]^2 + 2m[0,2]m[1,1] + 2m[0,2] - 2m[1,1], m[0,1]m[1,0]
- m[0,2]m[1,0] - m[1,1] + m[1,2], -2m[0,2]m[1,0] + 2m[1,2],
m[1,0]m[1,1] + m[0,2]m[1,2] - m[1,1]m[1,2] - m[1,2], -m[1,1]^2
+ m[1,2]^2, m[0,2]m[1,1] + m[1,0]m[1,2] - m[1,2]^2 - m[1,1],
-m[1,0]^2m[1,2] + m[1,0]m[1,2]^2 - m[0,2]m[1,2] + m[1,2],
m[0,1]m[1,2] + m[1,0]m[1,2] - m[1,2]^2 - m[1,1], -m[0,1]m[1,1]
+ m[0,2]m[1,2]]
-------------------------------
```

We perform some computations using the independence ideal. The elimination ideal $I \cap \mathbb{R}[m_{1,0}, m_{0,1}]$ is computed, then the following moments are reduced to their normal form: $E(X_1^4 X_2^5) = p_2 - p_5$, $E(X_2^5) = p_2 + p_3 - p_5 - p_6$ and $E(X_1^4) = p_1 + p_2 + p_4 + p_5$. We also reduce products of moments, e.g. $E(X_2^5)E(X_1^4) = (p_2 + p_3 - p_5 - p_6)(p_1 + p_2 + p_4 + p_5)$.

```
Elim(m[0,0],Elim(m[1,1],Elim(m[1,2],Elim(m[0,2],I))));
NF(P2-P5,I);
NF(P2+P3-P5-P6,I);
NF(P1+P2+P4+P5,I);
NF((P2+P3-P5-P6)*(P1+P2+P4+P5),I);
```

Problem 19: Fourth CoCoAcode.

```
Ideal(-m[0,1]^3m[1,0]^3 + m[0,1]^3m[1,0] + m[0,1]m[1,0]^3
- m[0,1]m[1,0])
-------------------------------
m[1,2]
-------------------------------
m[0,1]
-------------------------------
-m[0,2] + m[1,1] + 1
-------------------------------
m[1,2]
-------------------------------
```

Note that $I \cap \mathbb{R}[m_{1,0}, m_{0,1}] = \langle m_{1,0}m_{0,1}(m_{1,0}^2 - 1)(m_{0,1}^2 - 1)\rangle$.

Problem 20 (Contingency table)

The ideals I, J, $R = \langle p_{0,1} + p_{1,0} + p_{0,0} + p_{1,1} - 1\rangle$ and $M = \langle p_{0,1}p_{1,0}p_{0,0}p_{1,1}\rangle$ are first defined. R and M specify side conditions: R that the probabilities sum to one and M will be used (in saturation) to rule out all zero probabilities.

```
Use T::=Q[p[0..1,0..1]];
I:=Ideal(p[0,1]p[1,0] - p[0,0]p[1,1]);
C1:=p[0,0]-(p[0,0]+p[1,0])*(p[0,0]+p[0,1]);
C2:=p[1,0]-(p[0,0]+p[1,0])*(p[1,0]+p[1,1]);
C3:=p[0,1]-(p[0,1]+p[1,1])*(p[0,0]+p[0,1]);
C4:=p[1,1]-(p[1,1]+p[0,1])*(p[1,1]+p[1,0]);
J:=Ideal(C1,C2,C3,C4);
R:=Ideal(p[0,1]+p[1,0]+p[0,0]+p[1,1]-1);
M:=Ideal(p[0,1]*p[1,0]*p[0,0]*p[1,1]);
I=J;
```

Problem 20: CoCoAcode for definitions.

```
-------------------------------
FALSE
-------------------------------
```

Clearly, $I \neq J$. We then explore what relationship I and J may have. The first check is for inclusion. We then explore the role with the restrictions R and M.

```
Intersection(I,J);
I+R;
J+R;
I+R=J+R;
Saturation(J,M);
Saturation(J,M)=I+R;
```

Problem 20: Second CoCoAcode.

```
Ideal(p[0,1]p[1,0] - p[0,0]p[1,1])
-------------------------------
Ideal(-p[0,1]p[1,0] + p[0,0]p[1,1], p[0,0] + p[0,1] + p[1,0] + p[1,1] - 1)
-------------------------------
Ideal(-p[0,0]^2 - p[0,0]p[0,1] - p[0,0]p[1,0] - p[0,1]p[1,0] + p[0,0],
 -p[0,0]p[1,0] - p[1,0]^2 - p[0,0]p[1,1] - p[1,0]p[1,1] + p[1,0],
 -p[0,0]p[0,1] - p[0,1]^2 - p[0,0]p[1,1] - p[0,1]p[1,1] + p[0,1],
 -p[0,1]p[1,0] - p[0,1]p[1,1] - p[1,0]p[1,1] - p[1,1]^2 + p[1,1],
 p[0,0] + p[0,1] + p[1,0] + p[1,1] - 1)
-------------------------------
TRUE
-------------------------------
Ideal(p[0,0] + p[0,1] + p[1,0] + p[1,1] - 1,
p[0,1]p[1,0] + p[0,1]p[1,1] + p[1,0]p[1,1] + p[1,1]^2 - p[1,1])
-------------------------------
TRUE
-------------------------------
```

We verify that $I \cap J = I$ and therefore $I \subset J$. It then turns out that $I + R = J + R$, i.e. adding the restriction of sum to one makes the two ideals the same. Why the generators are different in this step?

Then the saturation $J : M^{\infty}$ is performed. It turns out that the result $J + R$ could be achieved by saturating J with M.

Problem 21 (Monomial ideal)

```
Use T::=Q[x[1..2]];
I:=Ideal(x[1]^3*x[2],x[1]*x[2]^2,x[2]^4);
J:=Ideal(x[1]^3*x[2],x[1]*x[2]^2,x[2]^4,x[1]^3*x[2]^4);
ReducedGBasis(I);
ReducedGBasis(J);
```

Problem 21: CoCoAcode for 21a.

```
[x[1]^3x[2], x[1]x[2]^2, x[2]^4]
-------------------------------
[x[1]^3x[2], x[1]x[2]^2, x[2]^4]
-------------------------------
```

The equality between I and J is verified by the equality of their Gröbner basis. Then the Hilbert function and Hilbert series are constructed.

Recall that, for a quotient ring $\mathbb{R}[x]/I$, its Hilbert function $\mathrm{H}(t)$ counts the number of monomials of degree t that do not belong to the ideal of leading terms $\langle LT(I)\rangle$. When the degree t is sufficiently large, $\mathrm{H}(t)$ is a polynomial with rational coefficients called the Hilbert polynomial. The smallest integer t_0 such that for $t \geq t_0$, $\mathrm{H}(t)$ is the Hilbert polynomial is called the *index of regularity*. The power series $\mathrm{HS}(s) = \sum_{t=0}^{\infty} \mathrm{H}(t)s^t$ is called the Hilbert series. If $\mathrm{HS}(s)$ is the Hilbert series, a simple calculation shows that

$$\mathrm{H}(t) = \lim_{s\to 0} \frac{1}{t!} \frac{\partial^t \mathrm{HS}(s)}{\partial s^t}. \tag{22}$$

The above result is easily seen by writing the Hilbert series in the equivalent form

$$\mathrm{HS}(s) = \sum_{\alpha\in L} s^{\sum_{i=1}^{d} \alpha_i}.$$

```
Hilbert(T/I);
HilbertSeries(T/I);
```

Problem 21: CoCoAcode for 21b.

```
H(0) = 1
H(1) = 2
H(2) = 3
H(3) = 3
H(t) = 1   for t >= 4
-------------------------------
(1 + x[1] + x[1]^2 - 2x[1]^4) / (1-x[1])
-------------------------------
```

For this example, the index of regularity is $t_0 = 4$, and the Hilbert polynomial is 1. The Hilbert series is

$$\begin{aligned}\mathrm{HS}(s) &= \sum_{t=0}^{\infty} \mathrm{H}(t) s^t = 1 + 2s + 3s^2 + 3s^3 + s^4 + s^5 + s^6 + \ldots \\ &= 1 + 2s + 3s^2 + 3s^3 + \frac{s^4}{1-s} = \frac{1 + s + s^2 - 2s^4}{1-s}.\end{aligned}$$

An example of computing the Hilbert function from the Hilbert series is

$$\mathrm{H}(3) = \lim_{s\to 0} \frac{1}{3!} \frac{\partial^3 \mathrm{HS}(s)}{\partial s^3} = \lim_{s\to 0} \frac{1}{6} \left(\frac{18 - 48s + 72s^2 - 48s^3 + 12s^4}{(1-s)^4} \right) = \frac{18}{6} = 3.$$

What is the interpretation of this result?

Problem 22 (Monomial ideal)

```
Use T::=Q[x[1..2]];
D:=[[0,0],[1,0],[0,1],[1,1],[2,0],[0,2]];
I:=IdealOfPoints(D);
LT(I);
Hilbert(T/I);
HilbertSeries(T/I);
```

Problem 22: CoCoAcode.

```
Ideal(x[2]^3, x[1]x[2]^2, x[1]^2x[2], x[1]^3)
-------------------------------
WARNING! HilbertPoincare input not homogeneous: computing LT...
H(0) = 1
H(1) = 2
H(2) = 3
H(t) = 0   for t >= 3
-------------------------------
WARNING! HilbertPoincare input not homogeneous: computing LT...
(1 + 2x[1] + 3x[1]^2)
-------------------------------
```

The index of regularity is $t_0 = 3$ and the Hilbert polynomial is zero. How do you interpret the obtained Hilbert function? What is the result and interpretation of the following limit

$$\lim_{s \to 1} \mathrm{HS}(s)?$$

Can you do this limit for Problem 21?

Problem 23 (Monomial ideal)

The initial step consists in listing sets of elements that cut off the system from working. A list of these sets is $\{12, 45, 235, 134, 1245\}$, which are used to construct the ideal I.

```
Use R::=Q[x[1..5]];
I:=Ideal(x[1]*x[2],x[4]*x[5],x[2]*x[3]*x[5],x[1]*x[3]*x[4],
x[1]*x[2]*x[4]*x[5]);
GBasis(I);
```

Problem 23: CoCoAcode for 23a.

The Gröbner basis of I gives a list of minimal elements representing the failure of the network.

```
[x[1]x[2], x[4]x[5], x[2]x[3]x[5], x[1]x[3]x[4]]
-------------------------------
```

The next step involves the multigraded Hilbert series. Let W be a square matrix of size $d \times d$, with integer entries and positive entries in the first row

(note that full rank of W is not needed). The multigraded Hilbert Series of the quotient ring $\mathbb{R}[x]/I$ is defined as

$$\mathrm{HS}_W(x) = \sum_{\alpha \in L} x^{W^T \alpha}, \tag{23}$$

where L is the usual notation for the set of exponents of monomials which do not lie in the ideal of leading terms $\langle LT(I) \rangle$.

```
W:=Identity(5);
HilbertSeriesMultiDeg(R/Ideal(0),W);
HilbertSeriesMultiDeg(R/I,W);
```

Problem 23: CoCoAcode for 23b.

```
---  Non-simplified HilbertPoincare' Series  ---
(1) / ( (1-x[1]) (1-x[2]) (1-x[3]) (1-x[4]) (1-x[5]) )
-------------------------------
---  Non-simplified HilbertPoincare' Series  ---
( - 2x[1]x[2]x[3]x[4]x[5] + x[1]x[2]x[3]x[4] + x[1]x[2]x[3]x[5]
+ x[1]x[2]x[4]x[5] + x[1]x[3]x[4]x[5] + x[2]x[3]x[4]x[5]
- x[1]x[3]x[4] - x[2]x[3]x[5] - x[1]x[2] - x[4]x[5] + 1)
 / ( (1-x[1]) (1-x[2]) (1-x[3]) (1-x[4]) (1-x[5]) )
-------------------------------
```

Using identity matrix for W, the multigraded Hilbert series for $\mathbb{R}[x]/\langle 0 \rangle$ is

$$\frac{1}{(1-x_1)(1-x_2)(1-x_3)(1-x_4)(1-x_5)}$$

and corresponds to the sum over all monomials in five variables, that is $L = \mathbb{Z}^5$, i.e. the most extensive L that generates the whole ring. The multigraded Hilbert series of $\mathbb{R}[x]/I$ is

$$\mathrm{HS}_W(x) = \frac{\begin{pmatrix} 1 - x_1x_2 - x_4x_5 - x_1x_3x_4 - x_2x_3x_5 + x_1x_2x_3x_4 + x_1x_2x_3x_5 \\ +x_1x_2x_4x_5 + x_1x_3x_4x_5 + x_2x_3x_4x_5 - 2x_1x_2x_3x_4x_5 \end{pmatrix}}{(1-x_1)(1-x_2)(1-x_3)(1-x_4)(1-x_5)}$$

In the above expression the usual inclusion-exclusion sums can be read directly, and in turned used to construct Bonferroni bounds. Note that the set partition $\mathbb{Z}^5 = L \cup \{\mathbb{Z}^5 \setminus L\}$ is reflected in the multigraded Hilbert series constructed for each set, i.e. $\mathrm{HS}(\mathbb{R}[x]/\langle 0 \rangle) = \mathrm{HS}(\mathbb{R}[x]/I) + \mathrm{HS}(I)$.

The last step points to the relation between Hilbert series and multigraded Hilbert series. If we consider a W matrix with unit elements in the first row and zeros elsewhere, then both series coincide.

```
W:=Mat([[1,1,1,1,1]]);
HilbertSeriesMultiDeg(R/I,W);
HilbertSeries(R/I);
Hilbert(R/I);
Hilbert(R/Ideal(0));
```

Problem 23: CoCoAcode for interpreting $\mathrm{HS}(s)$ and $\mathrm{HS}_W(x)$.

```
1 + 2x[1] + x[1]^2 - 2x[1]^3) / (1-x[1])^3
-------------------------------
(1 + 2x[1] + x[1]^2 - 2x[1]^3) / (1-x[1])^3
-------------------------------
H(0) = 1
H(t) = t^2 + 5t - 1   for t >= 1
-------------------------------
H(t) = 1/24t^4 + 5/12t^3 + 35/24t^2 + 25/12t + 1   for t >= 0
-------------------------------
```

The Hilbert polynomial $t^2 + 5t - 1$ satisfies Equation (22). Note that the Hilbert function for $\mathbb{R}[x]/\langle 0\rangle$ has the combinatorial structure

$$\mathrm{H}(t) = \frac{1}{24}t^4 + \frac{5}{12}t^3 + \frac{35}{24}t^2 + \frac{25}{12}t + 1 = \binom{t+4}{4}.$$

This formula counts the number of elements of degree t in $\mathbb{Z}^5$.

Problem 24 (Monomial ideal)

A minimal cut set for the network is used to generate the required ideal: $I = \langle a, g, ef, bcf, bcd, cde\rangle$. Then a multigraded Hilbert series is constructed. We use the identity of size 7 for the weights W. The Hilbert function and Hilbert series for the quotient ring are also computed.

```
Use T::=Q[a,b,c,d,e,f,g];
I:=Ideal(a,g,e*f,b*c*f,b*c*d,c*d*e);
HilbertSeries(T/I);
Hilbert(T/I);
W:=Identity(7);
HilbertSeriesMultiDeg(T/I,W);
```

Problem 24: CoCoAcode.

```
(1 + 2a + 2a^2 - a^3) / (1-a)^3
-------------------------------
H(0) = 1
H(t) = 2t^2 + 3t   for t >= 1
-------------------------------
---  Non-simplified HilbertPoincare' Series  ---
( - abcdefg + abcdef + abcdeg + abcdfg + abcefg + acdefg + bcdefg
- abcde - abcdf - abcef - acdef - bcdef - abcdg - acdeg - bcdeg
- abcfg - bcdfg - bcefg - cdefg + abcd + acde + bcde + abcf + bcdf
+ bcef + cdef + bcdg + cdeg + bcfg - aefg - bcd - cde - bcf + aef
+ efg - ef + ag - a - g + 1) / ( (1-a) (1-b) (1-c) (1-d) (1-e)
(1-f) (1-g) )
-------------------------------
```

It is important to remember that CoCoAdoes not simplify multigraded Hilbert series. In this example, further simplification of HS_W is possible:

$$
\begin{aligned}
\mathrm{HS}_W &= \frac{\begin{pmatrix} 1-a-g-ef+ag-bcd-cde-bcf+aef+efg \\ +abcd+acde+bcde+abcf+bcdf+bcef+cdef \\ +bcdg+cdeg+bcfg-aefg-abcfg-bcdfg-bcefg \\ -cdefg-acdef-bcdef-abcdg-acdeg-bcdeg \\ -abcde-abcdf-abcef-abcdefg+abcdef+abcdeg \\ +abcdfg+abcefg+acdefg+bcdefg \end{pmatrix}}{(1-a)(1-b)(1-c)(1-d)(1-e)(1-f)(1-g)} \\
&= \frac{1-ef-bcd-cde-bcf+bcde+bcdf+bcef+cdef-bcdef}{(1-b)(1-c)(1-d)(1-e)(1-f)}
\end{aligned}
$$

How do you interpret this result?

Problem 25 (Monomial ideal)

The matrices used for the computations are $W_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, $W_2 = \begin{pmatrix} 5 & 3 \\ -1 & 1 \end{pmatrix}$ and $W_3 = \begin{pmatrix} 1 & 1 \\ 3 & 2 \end{pmatrix}$.

```
Use T::=Q[x[1..2]];
D:=[[0,0],[1,0],[0,1],[1,1],[2,0],[0,2]];
I:=IdealOfPoints(D);
W1:=Identity(2);
W2:=Mat([[5,3],[-1,1]]);
W3:=Mat([[1,1],[3,2]]);
QuotientBasis(I);
HilbertSeriesMultiDeg(T/I,W1);
HilbertSeriesMultiDeg(T/I,W2);
HilbertSeriesMultiDeg(T/I,W3);
```

Problem 25: CoCoAcode.

```
[1, x[2], x[2]^2, x[1], x[1]x[2], x[1]^2]
-------------------------------
---  Non-simplified HilbertPoincare' Series  ---
(x[1]^3x[2] + x[1]^2x[2]^2 + x[1]x[2]^3 - x[1]^3 - x[1]^2x[2]
- x[1]x[2]^2 - x[2]^3 + 1) / ( (1-x[1]) (1-x[2]) )
-------------------------------
WARNING! HilbertPoincare input not homogeneous: computing LT...
---  Non-simplified HilbertPoincare' Series  ---
(x[1]^18/x[2]^2 + x[1]^16 + x[1]^14x[2]^2 - x[1]^15/x[2]^3
- x[1]^13/x[2] - x[1]^11x[2] - x[1]^9x[2]^3 + 1) / ( (1-x[1]^5/x[2])
(1-x[1]^3x[2]) )
-------------------------------
WARNING! HilbertPoincare input not homogeneous: computing LT...
---  Non-simplified HilbertPoincare' Series  ---
(x[1]^4x[2]^11 + x[1]^4x[2]^10 + x[1]^4x[2]^9 - x[1]^3x[2]^9
- x[1]^3x[2]^8 - x[1]^3x[2]^7 - x[1]^3x[2]^6 + 1) / ( (1-x[1]x[2]^3)
(1-x[1]x[2]^2) )
-------------------------------
```

The basic ingredient to computating the multigraded Hilbert series is the basis for $\mathbb{R}[x]/I$, which are the monomials $\{1, x_1, x_2, x_1^2, x_1x_2, x_2^2\}$, in turn represented by the set of exponents $L = \{(0,0),(1,0),(0,1),(2,0),(1,1),(0,2)\}$.

Recall that the multigraded Hilbert series is the sum over the exponent set $\{W^T\alpha^T \text{ for } \alpha \in L\}$. We now examine each multigraded Hilbert series according to the weight matrix used.

$\mathbf{HS}_{W_1}$ As W_1 is the identity, the is the sum over L and

$$\mathrm{HS}_{W_1}(x) = 1 + x_1 + x_2 + {x_2}^2 + x_1x_2 + {x_1}^2$$

HS$_{W_2}$ Now the sum is over the set $\{(0,0),(5,-1),(3,1),(10,-2),(8,0),(6,2)\}$ with result

$$\mathrm{HS}_{W_2}(x) = 1 + \frac{{x_1}^5}{x_2} + {x_1}^3 x_2 + \frac{{x_1}^{10}}{{x_2}^2} + {x_1}^8 + {x_1}^6 {x_2}^2$$

HS$_{W_3}$ The set for the sum is $\{(0,0),(1,3),(1,2),(2,6),(2,5),(2,4)\}$

$$\mathrm{HS}_{W_1}(x) = 1 + x_1 {x_2}^3 + x_1 {x_2}^2 + {x_1}^2 {x_2}^6 + {x_1}^2 {x_2}^5 + {x_1}^2 {x_2}^4$$

All the above Hilbert series coincide with CoCoAoutput, after simplification.

Another example is the Hilbert series for the quotient ring $\mathbb{R}[x]/\langle 0\rangle$ using the weight matrix W_2.

```
Use T::=Q[x[1..2]];
W2:=Mat([[5,3],[-1,1]]);
HilbertSeriesMultiDeg(T/Ideal(0),W2);
```

Problem 25: CoCoAcode.

```
---  Non-simplified HilbertPoincare' Series  ---
(1) / ( (1-x[1]^5/x[2]) (1-x[1]^3x[2]) )
-------------------------------
```

For this example, the set L is $L = \mathbb{Z}^2$ and thus

$$\mathrm{HS}_{W_2}(x) = \sum_{(i,j)\in\mathbb{Z}^2} x_1^{5i+3j} x_2^{j-i} = \sum_{(i,j)\in\mathbb{Z}^2} \left(\frac{x_1^5}{x_2}\right)^i (x_1^3 x_2)^j = \frac{x_2}{(x_2 - {x_1}^5)\,(1 - {x_1}^3 x_2)}$$

which coincides with CoCoAoutput.

Problem 26 (Markov basis)

The array representing the 3×3 table is

1	r_1	r_2	r_3	c_1	c_2	c_3	
1	1	0	0	1	0	0	$p_{0,0}$
1	1	0	0	0	1	0	$p_{0,1}$
1	1	0	0	0	0	1	$p_{0,2}$
1	0	1	0	1	0	0	$p_{1,0}$
1	0	1	0	0	1	0	$p_{1,1}$
1	0	1	0	0	0	1	$p_{1,2}$
1	0	0	1	1	0	0	$p_{2,0}$
1	0	0	1	0	1	0	$p_{2,1}$
1	0	0	1	0	0	1	$p_{2,2}$

The first step consists in computing the kernel of the matrix $X = (1, r_1, r_2, c_1, c_2)$.

```
Use T::=Q[p[0..2,0..2]];
X:=Mat([[1,1,0,1,0],[1,1,0,0,1],[1,1,0,0,0],[1,0,1,1,0],
[1,0,1,0,1],[1,0,1,0,0],[1,0,0,1,0],[1,0,0,0,1],[1,0,0,0,0]]);
B:=LinKer(Transposed(X));
B;
```

Problem 26: First CoCoAcode for 26a.

```
[[1, 0, -1, 0, 0, 0, -1, 0, 1], [0, 1, -1, 0, 0, 0, 0, -1, 1],
[0, 0, 0, 1, -1, 0, -1, 1, 0], [0, 0, 0, 0, -1, 1, 0, 1, -1]]
-------------------------------
```

The elements of the kernel B are then used to construct the ideal of independence. Saturation is required to remove probabilities equal to zero.

```
C:=Indets(); L:=[];
Foreach R In B Do
P:=1; M:=1;
Foreach J In 1..9 Do
If R[J]=1 Then P:=P*C[J] Elsif R[J]=-1 Then M:=M*C[J] Endif;
EndForeach;
L:=[L,P-M]; L:=Flatten(L);
EndForeach;
I:=Ideal(L);
I;
S:=Saturation(I,Ideal(Product(C)));
GBasis(S);
```

Problem 26: Second CoCoAcode for 26a.

```
Ideal(-p[0,2]p[2,0] + p[0,0]p[2,2], -p[0,2]p[2,1] + p[0,1]p[2,2],
-p[1,1]p[2,0] + p[1,0]p[2,1], p[1,2]p[2,1] - p[1,1]p[2,2])
-------------------------------
Ideal(p[0,1]p[1,0] - p[0,0]p[1,1], p[0,2]p[1,0] - p[0,0]p[1,2],
p[0,2]p[1,1] - p[0,1]p[1,2], p[0,1]p[2,0] - p[0,0]p[2,1],
p[0,2]p[2,0] - p[0,0]p[2,2], p[1,1]p[2,0] - p[1,0]p[2,1],
p[0,2]p[2,1] - p[0,1]p[2,2], p[1,2]p[2,0] - p[1,0]p[2,2],
p[1,2]p[2,1] - p[1,1]p[2,2])
-------------------------------
```

Importantly, note that the Gröbner basis of the saturation ideal consists of all the 2×2 minors of the 3×3 matrix with entries

$$\begin{pmatrix} p_{0,0} & p_{0,1} & p_{0,2} \\ p_{1,0} & p_{1,1} & p_{1,2} \\ p_{2,0} & p_{2,1} & p_{2,2} \end{pmatrix}$$

The corresponding matrices with integer entries form the Markov basis:

$$\mathrm{mb} = \left\{ \begin{pmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & -1 & 0 \\ 0 & 0 & 0 \\ -1 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 & -1 \\ -1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{pmatrix}, \right.$$
$$\left. \begin{pmatrix} 0 & 1 & -1 \\ 0 & 0 & 0 \\ 0 & -1 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 & -1 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & -1 \\ -1 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 1 & -1 & 0 \\ -1 & 1 & 0 \end{pmatrix} \right\}$$

For the problem 26b, this Markov basis is used as generator of a random walk among the space of matrices with non-negative integer entries and same marginals as the required table. The following code in `R` implements the function `metro`, a Metropolis-Hastings algorithm to generate a random walk.

```
#--------------------------------------------------------#
#   Metropolis-Hastings Algorithm, test for independence   #
#   Code by Anne Krampe & Sonja Kuhnt (TU Dortmund)       #
#--------------------------------------------------------#
metro <- function(data.table,chain.length)
{
h<-function(x) # log hypergeometric density
{ if (sum(x>=0)<9) dens <- -5000 else dens<- -sum(lgamma(x+1))
return(dens);
}
mb1<-matrix(c(1,0,-1,0,0,0,-1,0,1),nrow=3) # Markov-Basis mb
mb2<-matrix(c(1,0,-1,-1,0,1,0,0,0),nrow=3) # 3x3 Table
mb3<-matrix(c(1,-1,0,0,0,0,-1,1,0),nrow=3)
mb4<-matrix(c(1,-1,0,-1,1,0,0,0,0),nrow=3)
mb5<-matrix(c(0,0,0,0,1,-1,0,-1,1),nrow=3)
mb6<-matrix(c(0,0,0,1,0,-1,-1,0,1),nrow=3)
mb7<-matrix(c(0,0,0,1,-1,0,-1,1,0),nrow=3)
mb8<-matrix(c(0,1,-1,0,0,0,0,-1,1),nrow=3)
mb9<-matrix(c(0,1,-1,0,-1,1,0,0,0),nrow=3)
mb<-list(mb1,mb2,mb3,mb4,mb5,mb6,mb7,mb8,mb9)
chain<-vector(chain.length, mode="list")
chain[[1]] <- data.table # Metropolis-Hastings Algorithm
for(k in 2:chain.length)
{ candno <- sample(length(mb), 1)
vorz<-sample(c(-1,1), 1) # sign of move
ru <- runif(1,0,1)
cand <- mb[[candno]] * vorz # move + sign of move
datanew <- chain[[k-1]]+cand
alpha<-exp(h(datanew)-h(chain[[k-1]]))
chain[[k]] <- if(ru < min(alpha,1)) datanew else chain[[k-1]]
}
save(chain, file = "res.txt")
}
```

Problem 26b: `R` function `metro`.

The inputs to the function `metro` are the 3×3 contingency table and the desired number of steps in the random walk. In turn, the function stores the tables in the walk in a file called `"res.txt"`. The contains of this file are

loaded and processed using the R function chisq.test, that computes the χ^2 statistic for a contingency table. The data is contingency table (10) of Page 6.

```
dat<-matrix(nrow=3,ncol=3,byrow=TRUE,c(2,3,3,3,3,5,4,5,6))
chiobs<-chisq.test(dat)$statistic
total<-5000
metro(dat,total)
load("res.txt") # chain has the random walk
tab<-matrix(nrow=total,ncol=1)
kount<-1; pv<-0;
for(k in 1:total)
{ tab[k,1]<-chisq.test(chain[[k]])$statistic;
if (tab[k,1]>chiobs) pv<-pv+1 } hist(tab,breaks=35)
chiobs
pv/total
```

Problem 26b: R code for obtaining the p-value of independence hipothesis.

The observed statistic X^2 is stored in chiobs, while the p-value for the independence test is computed by pv/total.

```
>chiobs
X-squared
0.2473359
> pv/total
[1] 0.9864
```

The p-value obtained shows a very high agreement between the data and the null hypothesis. Figure 2 shows the histogram of the X^2 statistic of the 5000 tables generated with the Metropolis Hastings algorithm.

Alternatively the same R function chisq.test can be used directly to generate a version of the random walk.

```
> chisq.test(dat,simulate.p.value=TRUE,B=total)

        Pearson's Chi-squared test with simulated p-value (based on 5000
        replicates)

data:  dat
X-squared = 0.2473, df = NA, p-value = 1
```

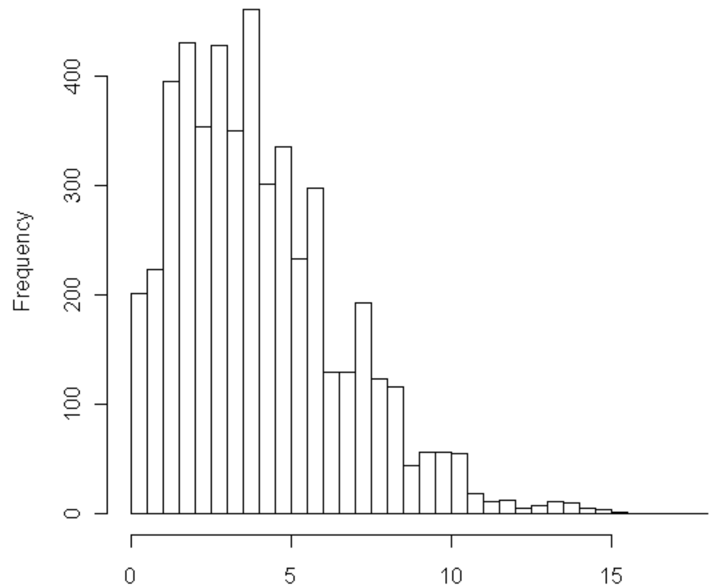


Figure 2: Histogram of results of Metropolis Hastings algorithm.

A comparison is made with a generalised linear model fitted to the same table. This is done with the following `R` code.

```
counts<-data.frame(
y=c(2,3,3,3,3,5,4,5,6),
ro=c(1,1,1,2,2,2,3,3,3),
co=c(1,2,3,1,2,3,1,2,3))
summary(glm(y ro+co,data=counts,family=poisson))
```

Problem 26c: `R` code for GLM.

```
Call:
glm(formula = y ~ ro + co, family = poisson, data = counts)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.32423  -0.07235   0.02930   0.07031   0.23483

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.2076     0.6806   0.305    0.760
ro            0.3139     0.2152   1.458    0.145
co            0.2224     0.2126   1.046    0.296

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 3.53827  on 8  degrees of freedom
Residual deviance: 0.25155  on 6  degrees of freedom
AIC: 34.717

Number of Fisher Scoring iterations: 4
```

The p-value using the GLM is obtained with the residual deviance. The formula to evaluate is $\Pr(\chi^2_6 \geq 0.25155)$.

```
> pchisq(q=0.25155,df=6,lower.tail=FALSE)
[1] 0.9996981
```

The p-values for all the tables in Problem 26 and three different methods (Markov basis 5000 steps MH, `chisq` simulation option 5000 steps and GLM) are summarized below.

Table number	10	11	12	13	14
Markov basis	0.9864	0.9772	0.3014	0.0000	0.6906
`chisq`	1.0000	0.9882	0.3001	0.0004	0.7694
GLM	0.9997	0.9887	0.4030	0.0001	0.8712

Problem 27 (Design of Experiments)

The design ideal is

$$\mathrm{Ideal}(\mathcal{D}) = \langle x_i(x_i - 1) : i = 1, \ldots, k \rangle.$$

The leading terms of the GBasis are $x_1^2, \ldots, x_k^2$ and thus the support of a hierarchical polynomial regression model is

$$\mathrm{SM}_{01} = \{x^\alpha : \alpha \in \mathcal{D}\}\,.$$

The set of polynomials $\{x_i(x_1-1) : i = 1, \ldots, k\}$ is a total Gbasis and thus Alg-Fan=Stat-Fan=SM_{01}.

The normal forms are

$$\mathrm{NormalForm}(x_1^2 x_2, \mathrm{Ideal}(\mathcal{D})) = x_1 x_2$$

and

$$\mathrm{NormalForm}(x_1^2 x_2^2, \mathrm{Ideal}(\mathcal{D})) = x_1 x_2.$$

The general formula is

$$\mathrm{NormalForm}(x^\alpha, \mathrm{Ideal}(\mathcal{D})) = x^\beta,$$

where $\beta = (1_{\alpha_1>0}, \ldots, 1_{\alpha_k>0})$.

Problem 28 (Design of Experiments)

The design coordinate 0 goes to -1 so we obtain $\mathcal{D}' = \{-1, 1\}^k$. Examples of such transformations are change of units, e.g. Celsius to Kelvin temperature scale.

A Gröbner basis for the design ideal is $\{x_i^2 - 1 : i = 1, \ldots, k\}$ and thus $\mathrm{SM}_{\pm 1} = \mathrm{SM}_{01}$ of Problem 27.

The alias table is changed to $X_1^2 = \ldots = X_k^2 = 1$ and therefore

$$\mathrm{NormalForm}(x^\alpha, \mathrm{Ideal}(\mathcal{D}')) = x^\beta$$

with $\beta = (\alpha_1 \mod 2, \ldots, \alpha_k \mod 2)$

Statistically, it might be relevant to express the same function in a different parametrization as it leads to interpretation of parameters. The validity of Gröbner methods is not affected as the division algorithm operates linearly on the coefficients (J. Peixoto, 1987, The American Statistician).

Problem 29 (Design of Experiments)

Plot the design and solve the systems of equations. All trivial.

Computing the algebraic fan of $\mathcal{D}$ is straightforward as the design is full factorial and Alg-Fan($\mathcal{D}$)=Stat-Fan=($\mathcal{D}$) = $\{1, a, b, c, ab, ac, bc, abc\}$

For the computation of the fan of $\mathcal{F}$, a family of universal ordering vectors is required, see for details [Babson et al., 1997]. For this example, the twelve weighing vectors obtained by the coordinate permutations of the vectors $(1, 2, 4), (2, 3, 4)$ is sufficient.

```
Permutations([1,2,4]);
Permutations([2,3,4]);
```

Problem 29: CoCoAcode for the family of universal ordering vectors.

```
[[1, 2, 4], [1, 4, 2], [2, 1, 4], [2, 4, 1], [4, 1, 2], [4, 2, 1]]
-------------------------------
[[2, 3, 4], [2, 4, 3], [3, 2, 4], [3, 4, 2], [4, 2, 3], [4, 3, 2]]
-------------------------------
```

The computation with weighing vector $(1, 2, 4)$ is performed with the following CoCoAcode.

```
M:=Identity(3); M[1]:=[1,2,4];
Use T::=Q[a,b,c], Ord(M);
I:=Ideal(a^2-1,b^2-1,c^2-1,abc-1);
QuotientBasis(I);
```

Problem 29: CoCoAcode for obtaining a model in the fan.

```
[1, b, a, ab]
-------------------------------
```

Repeating the above computation with other weighing vectors yields other models in the algebraic fan of $\mathcal{F}$.

```
[1, c, a, ac]
-------------------------------
[1, c, b, bc]
-------------------------------
[1, c, b, a]
-------------------------------
```

Finally, the fan is the union of all the models obtained

$$\text{Alg-Fan}(\mathcal{F}) = \text{Stat-Fan}(\mathcal{F}) = \{\{1, a, b, c\}, \{1, a, b, ab\}, \{1, a, c, ac\}, \{1, b, c, bc\}\}.$$

Alias table($\mathcal{D}$): $A^2 = B^2 = C^2 = I$

Alias table($\mathcal{F}$): $\begin{array}{l} I = ABC = A^2 = B^2 = C^2 \\ A = BC \\ B = AC \\ C = AB \end{array}$

Use the CoCoA function `IdealAndSeparatorsOfPoints` or solve a linear system of equations to obtain $\text{Ind}_{\mathcal{F}\subset\mathcal{D}} = (1 + abc)/2$.

Using the transformation $a^2 - 1$ becomes $a(a-1)$ and $abc - 1$ becomes $8abc - 4ab - 4ac - 4bc + 2a + 2b + 2c - 2$. The alias table is messy, but the algebraic and statistical fans are unchanged, why?

The indicator $\text{Ind}_{\mathcal{F}\subset\mathcal{D}} = (a + b + c) - 2(ab + ac + bc) + 4abc$.

Problem 30 (Design of Experiments)

See [Maruri-Aguilar et al., 2007] Section 4.1. The homogeneous component of a mixture design can be computed starting from the design points using the CoCoA macro `IdealOfProjectivePoints`. Then the degree s subset of a basis of the quotient space can be computed checking the leading terms as described in the paper mentioned above. Alternatively see the following CoCoAmacros.

```
Define Divides(Led,Comp);
//Compares two monomials, the first argument is "leadterm".
//Returns 1 if Comp is divisible by Led
Suma:=0; Difference:=Log(Comp)-Log(Led);
For I:=1 To NumIndets() Do If Difference[I]<0 Then
Suma:=Suma+1; EndIf; EndFor;
If IsZero(Suma) Then R:=1; Else R:=0; EndIf;
Return R;
EndDefine;
```

Problem 30: First CoCoAmacro for homogeneous model of degree s.

```
Define HomogeneousBasis(Id,S);
//To obtain a homogeneous basis for R[x]s/Is.
//Inputs are homogeneous Ideal Id and degree S
Le:=Gens(LT(Id)); C:=Support(DensePoly(S)); L:=[];
ForEach I In C Do
Total:=0; J:=1;
While Total=0 And J<=Len(Le) Do Total:=Total+Divides(Le[J],I);
J:=J+1; EndWhile;
If IsZero(Total) Then Append(L,I); EndIf;
EndForEach;
Return L;
EndDefine;
```

Problem 30: Second CoCoAmacro for homogeneous model of degree s.

The following are useful macros for generating classic mixture designs.

```
Define SimplexLattice(K,M);
//Simplex lattice Scheffe:  K factors M+1 levels
L:=(0..M); For I:=1 To (M+1) Do L[I]:=L[I]/M EndFor;
L:=Tuples(L,K);
ForEach I In L Do If Sum(I)<>1 Then L:=Diff(L,[I]); EndIf;
EndForEach;
Return L;
EndDefine;
```

Problem 30: CoCoAcode for Simplex lattice design.

```
Define GSimplexLattice(K,M);
G:=-1; L:=[];
ForEach I In 1..K Do;
P:=1; ForEach J In 0..M Do P:=P*(x[I]-J/M); EndForEach;
G:=G+x[I]; L:=Concat(L,[P]);
EndForEach;
L:=Concat(L,[G]);
Return L;
EndDefine;
```

Problem 30: CoCoAcode for generators if the simplex lattice design ideal.

Once a degree s support for a saturated linear model has been considered, then one can employ typical statistical techniques. For example use the `glm` command in R or Splus to compute a generalised linear fit to the date.

Problem 31 (Design of Experiments)

Non-saturated means that the X matrix is not full rank having more rows than columns. The list of points of the performed runs is given below

r	l	a	e
-1	0	1	1
-1	1	1	1
-1	0	1	0
0	1	1	0
0	-1	1	0
1	0	1	0
1	0	1	-1
0	1	1	-1
0	0	1	-1
-1	-1	1	-1
-1	0	1	-1
-1	1	1	-1
1	-1	0	-1
1	0	0	-1
0	1	0	-1
-1	-1	0	-1
-1	0	0	-1
-1	1	0	-1
1	0	-1	-1
0	-1	-1	-1
0	0	-1	-1
-1	-1	-1	-1
-1	1	-1	-1

The ideal of the intended runs is generated by the polynomials $ea-e+a-1, a^3-a, e^3-e, l^3-l, r^3-r$ where $ea-e+a-1=(a-1)(e+1)$. A Gröbner basis for the perfomed experiment under the `DegRevLex` $(r \succ l \succ a \succ e)$ ordering is $ae+a-e-1$,
e^3-e, a^3-a, l^3-l, r^3-r,
re^2+re+e^2+e,
$rle+le^2+rl+le$,
$r^2e+l^2e-le^2+r^2+l^2-le-e-1$,

$r^2la + rla,$
$r^2l^2 + r^2l + rl^2 + rl,$
$l^2e^2 + l^2e - le^2 - le,$
$l^2a^2 - 5/2r^2l - 3/2rl^2 + 1/2r^2a + 1/2rla - l^2a + 1/2ra^2$
$+3/2l^2e - 1/2r^2 - 3/2rl - 1/2ra + 1/2la - 1/2a^2$
$-3/2le - 1/2l + 1/2,$
$rla^2 + 2r^2l + rl^2 - rla + la^2 - l^2e + rl - la + le,$
$r^2a^2 - r^2l - rl^2 + rla + l^2e - r^2 - rl + la - a^2 - le - l + 1,$
$rl^2a + 2r^2l + rl^2 - rla + l^2a - 2l^2e + rl - l^2 - la + 2le + l.$

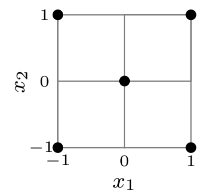
The degree 3 part of the 24 estimable models found by `DegRevLex` is shown in the next table.

	ea^2	ae^2	al^2	la^2	le^2	el^2	re^2	er^2	ra^2	ar^2	rl^2	lr^2	alr	ael	aer	elr
aelr			*	*				*	*	*	*	*	*			*
aerl			*	*		*			*	*	*	*	*			*
aler			*	*				*	*	*	*	*	*			*
alre			*	*	*			*	*	*	*	*	*			
arel			*	*	*	*			*	*	*	*	*			
arle			*	*	*	*			*	*	*	*	*			
ealr			*	*				*	*	*	*	*	*			*
earl			*	*		*			*	*	*	*	*			*
elar			*	*				*	*	*	*	*	*			*
elra			*	*				*	*	*	*	*	*			*
eral			*	*		*			*	*	*	*	*			*
erla			*	*		*			*	*	*	*	*			*
laer			*	*				*	*	*	*	*	*			*
lare			*	*	*			*	*	*	*	*	*			
lear			*	*				*	*	*	*	*	*			*
lera			*	*				*	*	*	*	*	*			*
lrae			*	*	*			*	*	*	*	*	*			
lrea			*	*	*			*	*	*	*	*	*			
rael			*	*	*	*			*	*	*	*	*			
rale			*	*	*	*			*	*	*	*	*			
real			*	*	*	*			*	*	*	*	*			
rela			*	*	*	*			*	*	*	*	*			
rlae			*	*	*	*			*	*	*	*	*			
rlea			*	*	*	*			*	*	*	*	*			

One could choose the set of terms common to the 24 initial orderings above and use it as a starting point for standard linear regression. For a full analysis see [Holliday et al., 1999].

Problem 32 (Design of Experiments)

The design is a cross design with points $(\pm 1, \pm 1)$ and $(0, 0)$ (five points in total).



For the computation of the algebraic fan of $\mathcal{D}$, a universal set of weighing vectors for designs with five points in two factors is

$$(5, 2), (7, 2), (5, 2), (5, 3), (7, 5), (5, 4)$$

together with its coordinate permutations, totalling twelve vectors. For the description of the set of ordering vectors, see [Babson et al., 1997] and [Maruri-Aguilar, 2006].

A similar computation to that of Problem 29 leads to the algebraic fan of $\mathcal{D}$, which has two leaves corresponding to models $1, x, y, xy, y^2$ and $1, x, y, xy, x^2$. The statistical fan equals the Alg-Fan. For central composite designs see Example 47 in [Pistone et al., 2000].

Problem 33 (Contingency table)

Let X and Y be binary random variables with levels 0 and 1. If X and Y are independent then the determinant of the contingency table is zero.

```
Use T::=Q[p[0..1,0..1]];
Indip:=[];
For I:=0 To 1 Do For J:=0 To 1 Do
Indip:=Concat( [p[I,J]-(p[I,0]+p[I,1])(p[0,J]+p[1,J])],
Indip);
EndFor; EndFor;
A:=p[0,0]p[1,1]-p[0,1]p[1,0];
NF(A,Cast(Indip,IDEAL));
```

Problem 33: First CoCoAcode.

```
0
-------------------------------
```

```
NF( p[1,1]-(p[1,0]+p[1,1])(p[0,1]+p[1,1]), Cast(A,IDEAL)) ;
NF( p[1,0]-(p[1,0]+p[1,1])(p[0,0]+p[1,0]), Cast(A,IDEAL)) ;
NF( p[0,1]-(p[0,0]+p[0,1])(p[0,1]+p[1,1]), Cast(A,IDEAL)) ;
NF( p[0,0]-(p[0,0]+p[0,1])(p[0,0]+p[1,0]), Cast(A,IDEAL)) ;
```

Problem 33: Second CoCoAcode.

```
0
-------------------------------
0
-------------------------------
0
-------------------------------
0
-------------------------------
```

Note that the computations do not change whatever the levels of x and Y are.

Problem 34 (Probability interpolation)

See the introductory chapter in [Pistone et al., 2000]. In the first case, $p(a, b, c) = 1/4$ (constant). Second instance, solve a system of linear equations to obtain $1/4 + 1/16a + 1/8b + 1/16c$.

Problem 35 (Probability interpolation)

(a) $\mathrm{E}_0(Y) = \sum_{x \in 2^{3-1}} Y(x)\,\mathrm{Prob}(x) = \sum_{(a,b,c) \in 2^{3-1}} (a + b + c)1/4 = 0.$

(b) $\mathrm{E}_0(Y^2) \sum_{x \in 2^{3-1}} Y(x)^2\,\mathrm{Prob}(x) = \sum_{(a,b,c) \in 2^{3-1}} (a+b+c)^2 1/4 = 12/4 = 3.$

(c) Obtain $y^2 - 2y - 3 = (y+1)(y-3)$. Y takes the two values in $\{-1, 3\} = \mathcal{D}^*$.

(d) As the sample space of the image of Y has cardinality two then a parametrization of a generic function over it has two degrees of freedom.

(e) It is the uniform probability over D^*.

(f) $\left\{ \begin{array}{cc} -1 & 3 \\ 3/4 & 1/4 \end{array} \right\}$. The system of equations $\theta_0 - \theta_1 = 3/4, \theta_0 + 3\theta_1 = 1/4$ gives $5/8 - 1/8y$.

(g) As a generic form of a density p_Y over D^* is $p_Y = \theta_0 + \theta_1 y$, then p_Y is known when the first two moments are known. We can use the formula $\mathrm{E}_*(P^*F) = \mathrm{E}_0(F)$ stating the equality between the expected value of a generic function PF w.r.t. the uniform distribution on the image space and the expected value of F w.r.t the probability P on the 2^{3-1} space. Here $F \in \{1, Y\}$ and $P = \theta_0 + \theta_1 y$. Thus,

$$\begin{array}{lll} F = 1 & \mathrm{E}_0(1) = 1 & \mathrm{E}_*((\theta_0 + \theta_1 Y)1) = \theta_0 + \theta_1 \mathrm{E}_*(Y) = \theta_0 + \theta_1 \\ F = Y & \mathrm{E}_0(Y) = 0 & \mathrm{E}_*((\theta_0 + \theta_1 Y)Y) = \mathrm{E}_*(\theta_0 Y + \theta_1 Y^2) \\ & & = \mathrm{E}_*(\theta_0 Y + \theta_1(2Y+3)) = \theta_0 + 5\theta_1 \end{array}$$

giving the system

$$1 = \theta_0 + \theta_1 \qquad 0 = \theta_0 + 5\theta_1$$

with solution $5/4 - Y/4$.

Problem 36 (Probability interpolation)

For the polynomial representation, solve $\theta_0 + \theta_1 a + \theta_2 b + \theta_3 c = p_{(a,b,c)}$ with $(a, b, c) \in 2^{3-1}$ to obtain a well-known formula

$$\frac{1}{4}\Big(\begin{array}{l} (p_{1,1,1} + p_{1,-1,-1} + p_{-1,1,-1} + p_{-1,-1,1}) \\ +(p_{1,1,1} + p_{1,-1,-1} - p_{-1,1,-1} - p_{-1,-1,1})a \\ +(p_{1,1,1} - p_{1,-1,-1} + p_{-1,1,-1} - p_{-1,-1,1})b \\ +(p_{1,1,1} - p_{1,-1,-1} - p_{-1,1,-1} + p_{-1,-1,1})c \end{array} \Big)$$

For the image probability, note that it takes value $v = p_{1,1,1}$ over 3 and $1 - p_{1,1,1}$ over -1. Solving $\theta_0 + 3\theta_1 = p_{1,1,1}, \theta_0 - \theta_1 = 1 - p_{1,1,1}$ gives $(3 - 2p_{1,1,1})/4 + y(2v-1)/4$.

For image with respect to uniform distribution, proceed as in (c)(iii). We need $\mathrm{E}\, P(Y) = \sum(a+b+c)p_{abc} = 3p_{1,1,1} - (p_{1,-1,-1} + p_{-1,1,-1} + p_{-1,-1,1})$. Thus the system to be solved becomes

$$1 = \theta_0 + \theta_1 \qquad 3p_{1,1,1} - (p_{1,-1,-1} + p_{-1,1,-1} + p_{-1,-1,1}) = \theta_0 + 5\theta_1$$

giving $(5-1v)/4 + (v-1)/4y$ with $v = 3p_{1,1,1} - (p_{1,-1,-1} + p_{-1,1,-1} + p_{-1,-1,1})$.

Problems 37, 38 and 39 (Conditional independence)

In [Meek et al., 2006] see Example 1; Example 5 and Examples 4 and 7.

Problem 40 (Conditional independence) [1]

```
Use T::=Q[p[0..2, 0..1,0..1, 0..1],e,m[0..2,0..1,0..1]], Lex;
-- fdty
--- Y indip F given (T,D)
M00:= Mat([ [ p[0,0,0,0], p[0,0,0,1] ], [ p[1,0,0,0],
p[1,0,0,1] ], [ p[2,0,0,0], p[2,0,0,1] ] ] );
M01:= Mat([ [ p[0,0,1,0], p[0,0,1,1] ], [ p[1,0,1,0],
p[1,0,1,1] ], [ p[2,0,1,0], p[2,0,1,1] ] ] );
M10:= Mat([ [ p[0,1,0,0], p[0,1,0,1] ], [ p[1,1,0,0],
p[1,1,0,1] ], [ p[2,1,0,0], p[2,1,0,1] ] ] );
M11:= Mat([ [ p[0,1,1,0], p[0,1,1,1] ], [ p[1,1,1,0],
p[1,1,1,1] ], [ p[2,1,1,0], p[2,1,1,1] ] ] );
Indipendence1:= Cast(Minors(2,M00), IDEAL) +
Cast(Minors(2,M01), IDEAL) + Cast(Minors(2,M10), IDEAL) +
Cast(Minors(2,M11), IDEAL) ;
Indipendence1;
```

Problem 40: CoCoAcode for (a).

```
Ideal(p[0,0,0,0]p[1,0,0,1] - p[0,0,0,1]p[1,0,0,0],
p[0,0,0,0]p[2,0,0,1] - p[0,0,0,1]p[2,0,0,0],
p[1,0,0,0]p[2,0,0,1] - p[1,0,0,1]p[2,0,0,0],
p[0,0,1,0]p[1,0,1,1] - p[0,0,1,1]p[1,0,1,0],
p[0,0,1,0]p[2,0,1,1] - p[0,0,1,1]p[2,0,1,0],
p[1,0,1,0]p[2,0,1,1] - p[1,0,1,1]p[2,0,1,0],
p[0,1,0,0]p[1,1,0,1] - p[0,1,0,1]p[1,1,0,0],
p[0,1,0,0]p[2,1,0,1] - p[0,1,0,1]p[2,1,0,0],
p[1,1,0,0]p[2,1,0,1] - p[1,1,0,1]p[2,1,0,0],
p[0,1,1,0]p[1,1,1,1] - p[0,1,1,1]p[1,1,1,0],
p[0,1,1,0]p[2,1,1,1] - p[0,1,1,1]p[2,1,1,0],
p[1,1,1,0]p[2,1,1,1] - p[1,1,1,1]p[2,1,1,0])
-------------------------------
```

[1] Thanks due to Geneletti and Dawid.

```
-- D indip F
N:= Mat( [ [ p[0,0,0,0]+p[0,0,0,1]+p[0,0,1,0]+p[0,0,1,1],
p[0,1,0,0]+p[0,1,0,1] +p[0,1,1,0]+p[0,1,1,1] ],
[p[1,0,0,0] +p[1,0,0,1]+p[1,0,1,0]+p[1,0,1,1],
p[1,1,0,0]+p[1,1,0,1]+p[1,1,1,0]+p[1,1,1,1] ],
[p[2,0,0,0]+p[2,0,0,1]+p[2,0,1,0]+p[2,0,1,1],
p[2,1,0,0]+p[2,1,0,1]+p[2,1,1,0]+p[2,1,1,1] ]] );
Indipendence2:=Cast(Minors(2,N), IDEAL);
Indipendence2;
```

Problem 40: CoCoAcode for (b).

```
Graph:=Indipendence1+Indipendence2;
Graph;
```

Problem 40: CoCoAcode for (c).

```
LogicalCond:=Ideal(
p[0,0,1,0], p[0,0,1,1], p[0,1,1,0], p[0,1,1,1],
p[1,0,0,0], p[1,0,0,1], p[1,1,0,0], p[1,1,0,1],
p[2,0,1,0], p[2,0,1,1],
p[2,1,0,0], p[2,1,0,1] );
Model:=LogicalCond+Graph;
Model=Ideal(1);
Model;
```

Problem 40: CoCoAcode for (d).

```
Manifest:=Ideal(
p[0,0,0,0]+p[0,1,0,0] - m[0,0,0],
p[0,0,0,1]+p[0,1,0,1] - m[0,0,1],
p[0,0,1,0]+p[0,1,1,0] - m[0,1,0],
p[0,0,1,1]+p[0,1,1,1] - m[0,1,1],
p[1,0,0,0]+p[1,1,0,0] - m[1,0,0],
p[1,0,0,1]+p[1,1,0,1] - m[1,0,1],
p[1,0,1,0]+p[1,1,1,0] - m[1,1,0],
p[1,0,1,1]+p[1,1,1,1] - m[1,1,1],
p[2,0,0,0]+p[2,1,0,0] - m[2,0,0],
p[2,0,0,1]+p[2,1,0,1] - m[2,0,1],
p[2,0,1,0]+p[2,1,1,0] - m[2,1,0],
p[2,0,1,1]+p[2,1,1,1] - m[2,1,1]);
SumToOne:= Ideal(
p[0,0,0,0]+p[0,0,0,1]+p[0,0,1,0]+p[0,0,1,1]
+ p[0,1,0,0]+p[0,1,0,1]+ p[0,1,1,0]+p[0,1,1,1] -1,
p[1,0,0,0]+p[1,0,0,1]+p[1,0,1,0]+p[1,0,1,1] +
p[1,1,0,0]+p[1,1,0,1]+ p[1,1,1,0]+p[1,1,1,1] -1,
p[2,0,0,0]+p[2,0,0,1]+p[2,0,1,0]+p[2,0,1,1] +
p[2,1,0,0]+p[2,1,0,1]+ p[2,1,1,0]+p[2,1,1,1] -1 );
Current:=Model+Manifest+SumToOne;
A:=Indets(); ElimC:=Current;
For I:=26 To 37 Do ElimC:=Elim(A[I],ElimC); EndFor;
ElimC;
```

Problem 40: CoCoAcode for (e).

The elimination ideal `ElimC` is

```
Ideal(-p[1,0,1,0] - p[1,0,1,1] + p[2,0,0,0] + p[2,0,0,1],
p[0,0,1,0], p[0,0,1,1], p[0,1,1,0], p[0,1,1,1], p[1,0,0,0],
p[1,0,0,1], p[1,1,0,0], p[1,1,0,1], p[2,0,1,0], p[2,0,1,1],
p[2,1,0,0], p[2,1,0,1], p[0,0,0,0] + p[0,0,0,1] + p[0,0,1,0]
+ p[0,0,1,1] + p[0,1,0,0] + p[0,1,0,1] + p[0,1,1,0] + p[0,1,1,1] - 1,
-p[0,1,0,0] - p[0,1,0,1] + p[1,1,1,0] + p[1,1,1,1],
-p[1,1,1,0] - p[1,1,1,1] - p[2,0,0,0] - p[2,0,0,1] + 1,
p[2,0,0,0] + p[2,0,0,1] + p[2,1,1,0] + p[2,1,1,1] - 1,
-p[0,0,0,1]p[2,1,1,0] - p[0,0,0,1]p[2,1,1,1] + p[0,0,0,1]
+ p[2,0,0,1]p[2,1,1,0] + p[2,0,0,1]p[2,1,1,1] - p[2,0,0,1],
p[0,0,0,1]p[1,1,1,1] - p[0,0,0,1]p[2,1,1,1] - p[1,1,1,1]p[2,0,0,1]
+ p[2,0,0,1]p[2,1,1,1], p[1,1,1,1]p[2,1,1,0] + p[1,1,1,1]p[2,1,1,1]
- p[2,1,1,0]p[2,1,1,1] - p[2,1,1,1]^2)
-------------------------------
```

```
Use T::=Q[ p[0..2, 0..1,0..1, 0..1],e,m[0..2,0..1,0..1]], Lex;
-- fdty
ETT:= e*(p[0,1,0,1]+p[0,1,0,0]) * (p[1,1,1,1]+p[1,1,1,0])
- ( p[1,1,1,1] * (p[0,1,0,1]+p[0,1,0,0])
- p[0,1,0,1] * (p[1,1,1,1]+p[1,1,1,0]) );
CurrentDouble:=Saturation(Current,
Ideal( (p[1,1,1,1]+p[1,1,1,0]) * (p[0,0,0,0]+p[0,0,0,1]) ) );
NF( ETT , CurrentDouble);
```

Problem 40: CoCoAcode for (f).

```
em[2,1,0]^2 + 2em[2,1,0]m[2,1,1] + em[2,1,1]^2
+ m[0,0,1]m[2,1,0] + m[0,0,1]m[2,1,1] - m[2,0,1]m[2,1,0]
- m[2,0,1]m[2,1,1] - m[2,1,0]m[2,1,1] - m[2,1,1]^2
-------------------------------
```

References

[Babson et al., 1997] Babson, E., Onn, S., and Thomas, R. (2003). The Hilbert zonotope and a polynomial time algorithm for universal Gröbner bases. *Adv. Appl. Math.* 30:529–544.

[Berstein et al., 2008] Berstein, Y., Maruri-Aguilar, H., Onn, S., Riccomagno, E., Wynn, H. (2008). Minimal average degree aberration and the state polytope for experimental designs. MUCM technical report 07/07. `arXiv:0808.3055`

[Consonni and Pistone, 2007] Consonni, G., and Pistone, G. (2007). Algebraic Bayesian analysis of contingency tables with possibly zero-probability cells. *Statist. Sinica* 17(4): 1355–1370.

[Cox et al., 1997] Cox, D., Little, J., and O'Shea, D. (1997). *Ideals, Varieties, and Algorithms*. Springer-Verlag, New York. Second Edition.

[Diaconis and Sturmfels, 1998] Diaconis, P., and Sturmfels, B. (1998). Algebraic algorithms for sampling from conditional distributions. *Ann. Statist.* 26(1):363–397.

[Drton et al., 2009] Drton, M., Sturmfels, B., and Sullivant, S. (2009). *Lectures on Algebraic Statistics*. Birkhäuser, Basel.

[Holliday et al., 1999] Holliday, T., Pistone, G., Riccomagno E., and Wynn, H. (1999). The Application of Computational Algebraic Geometry to the Analysis of Designed Experiments: a case study. *Computational Statistics* 14:2, 213–231.

[Hoşten et al., 2005] Hoşten, S., and Khetan, A., and Sturmfels, B. (2005). Solving the likelihood equations. *Found. Comput. Math.* 5(4):389–407.

[Maruri-Aguilar, 2006] Maruri-Aguilar, H. (2006). Universal Gröbner bases for designs of experiments. *Rend. Istit. Mat. Univ. Trieste* **37**(1-2): 95-119.

[Maruri-Aguilar et al., 2007] Maruri-Aguilar, H., Notari, R., and Riccomagno, E. (2007). On the description and identifiability analysis of mixture designs. *Statistica Sinica* **17**, 1417-1440.