```
> # This example is example 3 in Chapter 5 of the book Modelling population dynamics:  model
      formulation, fitting and assessment  using state−space methods
  # Press enter to activate each line of code (or use !!! button above to active all the code)
> restart;
> with(LinearAlgebra) :
> Dmat := proc(kappa, pars)
  local DD1, i, j;
  description "This proceedure form the derivative matrix given a vector of exhaustive summary
      terms, kappa, and a vector of parameters, pars";
  with(LinearAlgebra) :
  DD1 := Matrix(1 ..Dimension(pars), 1 ..Dimension(kappa)) :
  for i from 1 to Dimension(pars)  do
      for j from 1 to Dimension(kappa) do
          DD1[i, j] := diff(kappa[j], pars[i])
      end do
   end do;
  DD1;
   end proc:
> Estpars := proc(DD1, pars)
  local r, d, alphapre, alpha, PDE, FF, i, ans;
  description "Finds the estimable set of parameters for derivative matrix DD1 and vector of
      parameters pars";
  with(LinearAlgebra) :
  r := Rank(DD1);
  d := Dimension(pars) - r :
  alphapre := NullSpace(Transpose(DD1)) :
  α := Matrix(d, Dimension(pars)) : PDE := Vector(d) :
  FF := f(seq(pars[i], i = 1 ..Dimension(pars))) :
  for i from 1 to d do
         α[i, 1 ..Dimension(pars)] := alphapre[i] :
         PDE[i] := add(diff(FF, pars[j]) · α[i, j], j = 1 ..Dimension(pars)) :
   end do:
   ans := pdsolve({seq(PDE[i] = 0, i = 1 ..d)});
  end proc:
> Expan := proc(A, C, x0, n)
   local i, x, y, kappa, tt;
  description "Finds the exhaustive summary for the expansion method with n terms";
   y := eval(Multiply(C, x0), t = 0);
   x := eval(Multiply(A, x0), t = 1);
   tt := 1 :
   kappa := ⟨ ⟩ :
    for i from 1 to n do
         y := Multiply(eval(C, t = tt), x);
          tt := tt + 1 :
         x := Multiply(eval(A, t = tt), x);
         kappa := ⟨kappa, y⟩;
   end do:
   kappa := convert(kappa, Vector)
   end proc:
> PLUR := proc(D1)
```

local *pp*, *ll*, *u1*, *r1*, *DetU*;
description "This procedure finds a PLUR or turing factorisation of the matrix D1." :
$(pp, ll, u1, r1) := LUDecomposition(D1, output = ['P','L','U1','R']) :$
$DetU := Determinant(u1);$
$\langle DetU = 0, \{P = pp, L = ll, U = u1, R = r1\}\rangle :$
end proc:

> #The observation and transition matrices:

> $OO := \langle\langle 1|0\rangle, \langle 0|1\rangle\rangle$; $L := \langle\langle (1 - pii + \lambda \cdot pii)\cdot\phi_1 | \lambda\cdot\phi_2\rangle, \langle pii\cdot\phi_1|\phi_2\rangle\rangle$;

$$OO := \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$L := \begin{bmatrix} (pii\,\lambda - pii + 1)\,\phi_1 & \lambda\,\phi_2 \\ pii\,\phi_1 & \phi_2 \end{bmatrix} \quad \textbf{(1)}$$

> #The values of the state-equations at time zero are assumed to be the known constants:

> $ints := \langle n_{0,1}, n_{0,2}\rangle :$

> #The exhaustive summary is found using the proceedure Expan. In this case we include terms E $(\mathbf{y}_1),..., E(\mathbf{y}_4)$

> kappa := $Expan(L, OO, ints, 4)$;

$$\kappa := \Big[\big[(pii\,\lambda - pii + 1)\,\phi_1\, n_{0,1} + \lambda\,\phi_2\, n_{0,2}\big], \quad \textbf{(2)}$$
$$\big[pii\, n_{0,1}\,\phi_1 + n_{0,2}\,\phi_2\big],$$
$$\big[(pii\,\lambda - pii + 1)\,\phi_1\,((pii\,\lambda - pii + 1)\,\phi_1\, n_{0,1} + \lambda\,\phi_2\, n_{0,2}) + \lambda\,\phi_2\,(pii\, n_{0,1}\,\phi_1 + n_{0,2}\,\phi_2)$$
$$\big],$$
$$\big[pii\,\phi_1\,((pii\,\lambda - pii + 1)\,\phi_1\, n_{0,1} + \lambda\,\phi_2\, n_{0,2}) + \phi_2\,(pii\, n_{0,1}\,\phi_1 + n_{0,2}\,\phi_2)\big],$$
$$\big[(pii\,\lambda - pii + 1)\,\phi_1\,((pii\,\lambda - pii + 1)\,\phi_1\,((pii\,\lambda - pii + 1)\,\phi_1\, n_{0,1} + \lambda\,\phi_2\, n_{0,2})$$
$$+ \lambda\,\phi_2\,(pii\, n_{0,1}\,\phi_1 + n_{0,2}\,\phi_2)) + \lambda\,\phi_2\,(pii\,\phi_1\,((pii\,\lambda - pii + 1)\,\phi_1\, n_{0,1} + \lambda\,\phi_2\, n_{0,2})$$
$$+ \phi_2\,(pii\, n_{0,1}\,\phi_1 + n_{0,2}\,\phi_2))\big],$$
$$\big[pii\,\phi_1\,((pii\,\lambda - pii + 1)\,\phi_1\,((pii\,\lambda - pii + 1)\,\phi_1\, n_{0,1} + \lambda\,\phi_2\, n_{0,2}) + \lambda\,\phi_2\,(pii\, n_{0,1}\,\phi_1$$
$$+ n_{0,2}\,\phi_2)) + \phi_2\,(pii\,\phi_1\,((pii\,\lambda - pii + 1)\,\phi_1\, n_{0,1} + \lambda\,\phi_2\, n_{0,2}) + \phi_2\,(pii\, n_{0,1}\,\phi_1$$
$$+ n_{0,2}\,\phi_2))\big],$$
$$\big[(pii\,\lambda - pii + 1)\,\phi_1\,((pii\,\lambda - pii + 1)\,\phi_1\,((pii\,\lambda - pii + 1)\,\phi_1\,((pii\,\lambda - pii$$
$$+ 1)\,\phi_1\, n_{0,1} + \lambda\,\phi_2\, n_{0,2}) + \lambda\,\phi_2\,(pii\, n_{0,1}\,\phi_1 + n_{0,2}\,\phi_2)) + \lambda\,\phi_2\,(pii\,\phi_1\,((pii\,\lambda - pii$$
$$+ 1)\,\phi_1\, n_{0,1} + \lambda\,\phi_2\, n_{0,2}) + \phi_2\,(pii\, n_{0,1}\,\phi_1 + n_{0,2}\,\phi_2))) + \lambda\,\phi_2\,(pii\,\phi_1\,((pii\,\lambda - pii$$
$$+ 1)\,\phi_1\,((pii\,\lambda - pii + 1)\,\phi_1\, n_{0,1} + \lambda\,\phi_2\, n_{0,2}) + \lambda\,\phi_2\,(pii\, n_{0,1}\,\phi_1 + n_{0,2}\,\phi_2))$$
$$+ \phi_2\,(pii\,\phi_1\,((pii\,\lambda - pii + 1)\,\phi_1\, n_{0,1} + \lambda\,\phi_2\, n_{0,2}) + \phi_2\,(pii\, n_{0,1}\,\phi_1 + n_{0,2}\,\phi_2)))\big],$$

$$\left[pii\,\phi_1\left(\left(pii\,\lambda-pii+1\right)\phi_1\left(\left(pii\,\lambda-pii+1\right)\phi_1\left(\left(pii\,\lambda-pii+1\right)\phi_1\,n_{0,1}+\lambda\,\phi_2\,n_{0,2}\right)+\lambda\,\phi_2\left(pii\,n_{0,1}\,\phi_1+n_{0,2}\,\phi_2\right)\right)+\lambda\,\phi_2\left(pii\,\phi_1\left(\left(pii\,\lambda-pii+1\right)\phi_1\,n_{0,1}+\lambda\,\phi_2\,n_{0,2}\right)+\phi_2\left(pii\,n_{0,1}\,\phi_1+n_{0,2}\,\phi_2\right)\right)\right)+\phi_2\left(pii\,\phi_1\left(\left(pii\,\lambda-pii+1\right)\phi_1\left(\left(pii\,\lambda-pii+1\right)\phi_1\,n_{0,1}+\lambda\,\phi_2\,n_{0,2}\right)+\lambda\,\phi_2\left(pii\,n_{0,1}\,\phi_1+n_{0,2}\,\phi_2\right)\right)+\phi_2\left(pii\,\phi_1\left(\left(pii\,\lambda-pii+1\right)\phi_1\,n_{0,1}+\lambda\,\phi_2\,n_{0,2}\right)+\phi_2\left(pii\,n_{0,1}\,\phi_1+n_{0,2}\,\phi_2\right)\right)\right)\right]\right]$$

> $pars := \langle pii, \lambda, \phi_1, \phi_2 \rangle :$

> $D1 := Dmat(\text{kappa}, pars) : D1[1..4, 1..2]$

$$\begin{bmatrix} (\lambda-1)\,\phi_1\,n_{0,1} & n_{0,1}\,\phi_1 \\ pii\,n_{0,1}\,\phi_1+n_{0,2}\,\phi_2 & 0 \\ (pii\,\lambda-pii+1)\,n_{0,1} & pii\,n_{0,1} \\ \lambda\,n_{0,2} & n_{0,2} \end{bmatrix} \quad \textbf{(3)}$$

> $r := Rank(D1);\ d := Dimension(pars) - r;$

$$r := 4$$
$$d := 0 \quad \textbf{(4)}$$

> #This model is not parameter redundant and we can in theory estimate all the parameters.

> $PLURresults := PLUR(D1) : PLURresults[1]$

$$\left[\phi_1\left(pii\,\lambda\,n_{0,1}\,n_{0,2}\,\phi_1-pii\,n_{0,1}^2\,\phi_1-pii\,n_{0,1}\,n_{0,2}\,\phi_1+\lambda\,n_{0,2}^2\,\phi_2+n_{0,1}\,n_{0,2}\,\phi_1-n_{0,1}\,n_{0,2}\,\phi_2\right)^2\phi_2=0\right] \quad \textbf{(5)}$$

>