```
> # Inital code based on:
> # Example From Parameter Redundancy and Identifiability by Diana Cole
> # Section 9.2.2 :  Immigration integrated model
> # Additional exhaustive summaries considered for IPM Identifiability workshop, Bordeaux 24-28
      April
>
> restart;
> with(LinearAlgebra) :
> Dmat := proc(se, pars)
  local DD1, i, j;
  description "Form the derivative matrix";
  with(LinearAlgebra) :
  DD1 := Matrix(1 ..Dimension(pars), 1 ..Dimension(se)) :
  for i from 1 to Dimension(pars)  do
      for j from 1 to Dimension(se) do
          DD1[i, j] := diff(se[j], pars[i])
      end do
   end do;
  DD1;
   end proc:
> Estpar := proc(DD1, pars, ret)
  local r, d, alphapre, alpha, PDE, FF, i, j, ans;
  description "Finds the estimable set of parameters for derivative matrix DD1. If ret = 1 returns
      alpha, PDEs, estimable parameter combinations. Otherwise returns estimable parameter
      combinations";
  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:
   if ret = 1 then
           ans := ⟨pdsolve({seq(PDE[i] = 0, i = 1 ..d)}), {alpha}, {PDE}⟩ :
   else
          ans := pdsolve({seq(PDE[i] = 0, i = 1 ..d)}) :
   end if:
   ans :
  end proc:
> Ztransformys := proc(A, B, x0)
   local II, ys, kappa;
  description "Finds the z-transform function";
   II := IdentityMatrix(Dimension(A)[1]) :
   ys := simplify(Multiply(B, Multiply(MatrixInverse(z·II − A), Multiply(A, x0))));
  ys :
   end proc:
>
> caprecapexsum := proc(y, z, r, c)
```

```
local i, j, m,  P, aa, b, P1, P2, P1ans, P2ans;
 description "Creates the simpler exhaustive summary for the capture-recapture model";
with(LinearAlgebra) :
 P1 := Matrix(c, c) :
 P2 := Matrix(c, c) :
  if y = 1 then
      for i from 1 to c do
          for j from 1 to c do
                 aa[i, j] := phi :
          end do:
      end do:
 elif y = 2 then
       for i from 1 to c do
          for j from 1 to c do
                 aa[i, j] := phi[j] :
          end do:
      end do:
 elif y = 3 then
       for i from 1 to c do
          for j from 1 to c do
                 aa[i, j] := phi[i] :
          end do:
      end do:
 else
      for i from 1 to c do
          for j from 1 to c do
                 aa[i, j] := phi[i, j] :
          end do:
      end do:
 end if:
  if z = 1 then
      for i from 1 to (c + 1) do
          for j from 1 to (c + 1) do
                 b[i, j] := p :
          end do:
      end do:
 elif z = 2 then
       for i from 1 to (c + 1) do
          for j from 1 to (c + 1) do
                 b[i, j] := p[j] :
          end do:
      end do:
 elif z = 3 then
       for i from 1 to (c + 1) do
          for j from 1 to (c + 1) do
                 b[i, j] := p[i] :
          end do:
      end do:
else
      for i from 1 to (c + 1) do
          for j from 1 to (c + 1) do
                 b[i, j] := p[i, j] :
```

```
            end do:
        end do:
    end if:

    for i from 1 to Dimension(P1)[1] do
        for j from i to Dimension(P1)[2] do
            P1[i, j] := (aa[i, c + i - j])·b[i + 1, c + i + 1 - j];
        end do:
    end do:
    P1ans := P1[.., c - r + 1 ..c];
    for i from 1 to Dimension(P2)[1]do
        for j from i to Dimension(P2)[2] do
            P2[i, j] := (aa[i - 1, c + i - 1 - j])·(1- b[i, c + i - j]);
        end do:
    end do:
    P2ans := P2[2 ..c, c - r + 1 ..c];
    P := ⟨P1ans, P2ans⟩
    end proc:
> Matvec :=proc(P)
   local sizekappa, i, j, kappa, kappaindex;
  description "Converts the data matrix into a vector of non-zero terms";
  with(LinearAlgebra) :
  sizekappa := 0 :
  for i from 1 to Dimension(P)[1] do
      for j from 1 to Dimension(P)[2] do
          if (P[i, j] ≠ 0) then
            sizekappa := sizekappa + 1 :
          end if
      end do
   end do;
  κ := Vector(sizekappa) : kappaindex := 0 :
  for i from 1 to Dimension(P)[1] do
      for j from 1 to Dimension(P)[2] do
          if (P[i, j] ≠ 0) then
            kappaindex :=  kappaindex + 1 :
            κ[kappaindex] := P[i, j] :
          end if:
      end do:
   end do:
   κ;
  end proc:
>
> # Exhaustive summary for the census data
```

> $B := \langle 1|1\rangle; A := \left\langle \left\langle \frac{\text{phi}[1]\cdot\text{rho}}{2} \middle| \frac{\text{phi}[1]\cdot\text{rho}}{2} \right\rangle, \langle \text{phi}[a] + im|\text{phi}[a] + \text{im}\rangle \right\rangle; x0 := \langle \text{x}[0, 1], x[0, 2]\rangle :$

$$B := \begin{bmatrix} 1 & 1 \end{bmatrix} \quad \textbf{(1)}$$

$$A := \begin{bmatrix} \frac{\phi_1 \rho}{2} & \frac{\phi_1 \rho}{2} \\ \phi_a + im & \phi_a + im \end{bmatrix} \tag{1}$$

> $ys := Ztransformys(A, B, x0);$

$$ys := \frac{(x_{0,1} + x_{0,2})\,(\phi_1 \rho + 2\,im + 2\,\phi_a)}{-\phi_1 \rho - 2\,im + 2\,z - 2\,\phi_a} \tag{2}$$

> $kappa1 := Vector(2):$
> $kappa1[1] := coeff(numer(ys), z, 0) : kappa1[2] := coeff(denom(ys), z, 0) : kappa1 := kappa1;$

$$\kappa 1 := \begin{bmatrix} -(x_{0,1} + x_{0,2})\,(\phi_1 \rho + 2\,im + 2\,\phi_a) \\ \phi_1 \rho + 2\,im + 2\,\phi_a \end{bmatrix} \tag{3}$$

> $pars1 := \langle \mathrm{phi}[1], \mathrm{phi}[a], im, \mathrm{rho} \rangle :$

> $D1 := Dmat(kappa1, pars1);$

$$D1 := \begin{bmatrix} -(x_{0,1} + x_{0,2})\,\rho & \rho \\ -2\,x_{0,1} - 2\,x_{0,2} & 2 \\ -2\,x_{0,1} - 2\,x_{0,2} & 2 \\ -(x_{0,1} + x_{0,2})\,\phi_1 & \phi_1 \end{bmatrix} \tag{4}$$

> $nopars := Dimension(pars1); r := Rank(D1); d := Dimension(pars1) - r;$

$$nopars := 4$$
$$r := 1$$
$$d := 3 \tag{5}$$

> $Estpar(D1, pars1, 0);$

$$\{f(\phi_1, \phi_a, im, \rho) = \_F1(\phi_1 \rho + 2\,im + 2\,\phi_a)\} \tag{6}$$

>

>

> # kappa2 is an exhaustive summary for the adult capture-recapture data, and kappa3 is for the juvenile capture-recapture

> $P := caprecapexsum(1, 1, 2, 2):$

> $kappa2 := eval(Matvec(P), \{\mathrm{phi} = \mathrm{phi}[a], \mathrm{lambda} = p\});$

$$\kappa 2 := \begin{bmatrix} \phi_a p \\ \phi_a p \\ \phi_a p \\ \phi_a (1 - p) \end{bmatrix} \tag{7}$$

> # Simplified exhaustive summary by removing repeated terms

> $kappa2s := \langle kappa2[3], kappa2[4]\rangle;$

$$kappa2s := \begin{bmatrix} \phi_a\, p \\ \phi_a\,(1-p) \end{bmatrix} \qquad \textbf{(8)}$$

> $pars2 := \langle \mathrm{phi}[a], p\rangle :$

> $D2 := Dmat(kappa2s, pars2);$

$$D2 := \begin{bmatrix} p & 1-p \\ \phi_a & -\phi_a \end{bmatrix} \qquad \textbf{(9)}$$

> $nopars := Dimension(pars2); r := Rank(D2); d := Dimension(pars2) - r;$

$$nopars := 2$$
$$r := 2$$
$$d := 0 \qquad \textbf{(10)}$$

>

> $P := caprecapexsum(3, 1, 2, 2);$

$$P := \begin{bmatrix} \phi_1\, p & \phi_1\, p \\ 0 & \phi_2\, p \\ 0 & \phi_1\,(1-p) \end{bmatrix} \qquad \textbf{(11)}$$

> $kappa3 := eval(Matvec(P), \{seq(\mathrm{phi}[j] = \mathrm{phi}[a], j = 2..10), \mathrm{lambda} = p\});$

$$\kappa 3 := \begin{bmatrix} \phi_1\, p \\ \phi_1\, p \\ \phi_a\, p \\ \phi_1\,(1-p) \end{bmatrix} \qquad \textbf{(12)}$$

> # Simplified by removing repeated terms:

> kappa3s := ⟨kappa3[1], kappa3[3], kappa3[4]⟩;

$$kappa3s := \begin{bmatrix} \phi_1\, p \\ \phi_a\, p \\ \phi_1\,(1-p) \end{bmatrix} \qquad \textbf{(13)}$$

> $pars3 := \langle \mathrm{phi}[1], \mathrm{phi}[a], p\rangle :$

> $D3 := Dmat(kappa3s, pars3);$

$$D3 := \begin{bmatrix} p & 0 & 1-p \\ 0 & p & 0 \\ \phi_1 & \phi_a & -\phi_1 \end{bmatrix} \qquad \textbf{(14)}$$

> $nopars := Dimension(pars3); r := Rank(D3); d := Dimension(pars3) - r;$

$$nopars := 3$$
$$r := 3$$
$$d := 0 \tag{15}$$

>

> # Exhaustive summary for productivity data

> $kappa4 := \langle \text{rho}\cdot R[t] \rangle;$

$$\kappa 4 := \begin{bmatrix} \rho R_t \end{bmatrix} \tag{16}$$

> $pars4 := \langle \text{rho} \rangle$:

> $D4 := Dmat(kappa4, pars4);$

$$D4 := \begin{bmatrix} R_t \end{bmatrix} \tag{17}$$

> $nopars := Dimension(pars4); r := Rank(D4); d := Dimension(pars4) - r;$

$$nopars := 1$$
$$r := 1$$
$$d := 0 \tag{18}$$

>

> #Exhaustive summary for different combinations of data sets:

>

> $\text{kappa} := convert(\langle kappa1, kappa2s \rangle, Vector);$

$$\kappa := \begin{bmatrix} -\left(x_{0,1} + x_{0,2}\right)\left(\phi_1 \rho + 2\, im + 2\, \phi_a\right) \\ \phi_1 \rho + 2\, im + 2\, \phi_a \\ \phi_a p \\ \phi_a (1 - p) \end{bmatrix} \tag{19}$$

> $pars := \langle \text{phi}[1], \text{phi}[a], im, \text{rho}, p \rangle$:

> $D1 := Dmat(\text{kappa}, pars)$

$$D1 := \begin{bmatrix} -\left(x_{0,1} + x_{0,2}\right)\rho & \rho & 0 & 0 \\ -2\, x_{0,1} - 2\, x_{0,2} & 2 & p & 1 - p \\ -2\, x_{0,1} - 2\, x_{0,2} & 2 & 0 & 0 \\ -\left(x_{0,1} + x_{0,2}\right)\phi_1 & \phi_1 & 0 & 0 \\ 0 & 0 & \phi_a & -\phi_a \end{bmatrix} \tag{20}$$

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

$$r := 3$$
$$d := 2 \tag{21}$$

> $Estpar(D1, pars, 0);$

$$\left\{ f\left(\phi_1, \phi_a, im, \rho, p\right) = \_F1\left(\phi_a, \phi_1 \rho + 2\, im, p\right) \right\} \tag{22}$$

> kappa := $convert(\langle kappa1, kappa3s\rangle, Vector)$;

$$\kappa := \begin{bmatrix} -(x_{0,1}+x_{0,2})\,(\phi_1\,\rho + 2\,im + 2\,\phi_a) \\ \phi_1\,\rho + 2\,im + 2\,\phi_a \\ \phi_1\,p \\ \phi_a\,p \\ \phi_1\,(1-p) \end{bmatrix} \tag{23}$$

> $pars := \langle \text{phi}[1], \text{phi}[a], im, \text{rho}, p\rangle$:

> $D1 := Dmat(\text{kappa}, pars)$

$$D1 := \begin{bmatrix} -(x_{0,1}+x_{0,2})\,\rho & \rho & p & 0 & 1-p \\ -2\,x_{0,1}-2\,x_{0,2} & 2 & 0 & p & 0 \\ -2\,x_{0,1}-2\,x_{0,2} & 2 & 0 & 0 & 0 \\ -(x_{0,1}+x_{0,2})\,\phi_1 & \phi_1 & 0 & 0 & 0 \\ 0 & 0 & \phi_1 & \phi_a & -\phi_1 \end{bmatrix} \tag{24}$$

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

$$r := 4$$
$$d := 1 \tag{25}$$

> $Estpar(D1, pars, 0);$

$$\left\{ f(\phi_1, \phi_a, im, \rho, p) = \_F1\left(\phi_1, \phi_a, \frac{\phi_1\,\rho + 2\,im}{\phi_1}, p\right) \right\} \tag{26}$$

> kappa := $convert(\langle kappa1, kappa4\rangle, Vector)$;

$$\kappa := \begin{bmatrix} -(x_{0,1}+x_{0,2})\,(\phi_1\,\rho + 2\,im + 2\,\phi_a) \\ \phi_1\,\rho + 2\,im + 2\,\phi_a \\ \rho\,R_t \end{bmatrix} \tag{27}$$

> $pars := \langle \text{phi}[1], \text{phi}[a], im, \text{rho}\rangle$:

> $D1 := Dmat(\text{kappa}, pars)$

$$D1 := \begin{bmatrix} -(x_{0,1}+x_{0,2})\,\rho & \rho & 0 \\ -2\,x_{0,1}-2\,x_{0,2} & 2 & 0 \\ -2\,x_{0,1}-2\,x_{0,2} & 2 & 0 \\ -(x_{0,1}+x_{0,2})\,\phi_1 & \phi_1 & R_t \end{bmatrix} \tag{28}$$

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

$$r := 2$$

$$d := 2 \qquad (29)$$

> $Estpar(D1, pars, 0);$

$$\left\{ f\left(\phi_1, \phi_a, im, \rho\right) = \_F1\left(\rho, \frac{\phi_1 \rho}{2} + \phi_a + im\right) \right\} \qquad (30)$$

> $\text{kappa} := convert(\langle kappa1, kappa2s, kappa3s \rangle, Vector);$

$$\kappa := \begin{bmatrix} -\left(x_{0,1} + x_{0,2}\right)\left(\phi_1 \rho + 2\, im + 2\,\phi_a\right) \\ \phi_1 \rho + 2\, im + 2\,\phi_a \\ \phi_a p \\ \phi_a (1 - p) \\ \phi_1 p \\ \phi_a p \\ \phi_1 (1 - p) \end{bmatrix} \qquad (31)$$

> $pars := \langle \text{phi}[1], \text{phi}[a], im, \text{rho}, p \rangle :$

> $D1 := Dmat(\text{kappa}, pars)$

$$D1 := \begin{bmatrix} -\left(x_{0,1} + x_{0,2}\right)\rho & \rho & 0 & 0 & p & 0 & 1-p \\ -2\,x_{0,1} - 2\,x_{0,2} & 2 & p & 1-p & 0 & p & 0 \\ -2\,x_{0,1} - 2\,x_{0,2} & 2 & 0 & 0 & 0 & 0 & 0 \\ -\left(x_{0,1} + x_{0,2}\right)\phi_1 & \phi_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \phi_a & -\phi_a & \phi_1 & \phi_a & -\phi_1 \end{bmatrix} \qquad (32)$$

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

$$r := 4$$
$$d := 1 \qquad (33)$$

> $Estpar(D1, pars, 0);$

$$\left\{ f\left(\phi_1, \phi_a, im, \rho, p\right) = \_F1\left(\phi_1, \phi_a, \frac{\phi_1 \rho + 2\, im}{\phi_1}, p\right) \right\} \qquad (34)$$

> $\text{kappa} := convert(\langle kappa1, kappa3s, kappa4 \rangle, Vector);$

(35)

$$\kappa := \begin{bmatrix} -(x_{0,1}+x_{0,2})(\phi_1 \rho + 2\, im + 2\, \phi_a) \\ \phi_1 \rho + 2\, im + 2\, \phi_a \\ \phi_1 p \\ \phi_a p \\ \phi_1 (1-p) \\ \rho R_t \end{bmatrix} \tag{35}$$

> pars := ⟨phi[1], phi[a], im, rho, p⟩ :

> D1 := Dmat(kappa, pars)

$$D1 := \begin{bmatrix} -(x_{0,1}+x_{0,2})\rho & \rho & p & 0 & 1-p & 0 \\ -2x_{0,1}-2x_{0,2} & 2 & 0 & p & 0 & 0 \\ -2x_{0,1}-2x_{0,2} & 2 & 0 & 0 & 0 & 0 \\ -(x_{0,1}+x_{0,2})\phi_1 & \phi_1 & 0 & 0 & 0 & R_t \\ 0 & 0 & \phi_1 & \phi_a & -\phi_1 & 0 \end{bmatrix} \tag{36}$$

> r := Rank(D1); d := Dimension(pars) − r;

$$r := 5$$
$$d := 0 \tag{37}$$

> kappa := convert(⟨kappa1, kappa2s, kappa4⟩, Vector);

$$\kappa := \begin{bmatrix} -(x_{0,1}+x_{0,2})(\phi_1 \rho + 2\, im + 2\, \phi_a) \\ \phi_1 \rho + 2\, im + 2\, \phi_a \\ \phi_a p \\ \phi_a (1-p) \\ \rho R_t \end{bmatrix} \tag{38}$$

> pars := ⟨phi[1], phi[a], im, rho, p⟩ :

> D1 := Dmat(kappa, pars)

(39)

$$D1 := \begin{bmatrix} -\left(x_{0,1}+x_{0,2}\right)\rho & \rho & 0 & 0 & 0 \\ -2\,x_{0,1}-2\,x_{0,2} & 2 & p & 1-p & 0 \\ -2\,x_{0,1}-2\,x_{0,2} & 2 & 0 & 0 & 0 \\ -\left(x_{0,1}+x_{0,2}\right)\phi_1 & \phi_1 & 0 & 0 & R_t \\ 0 & 0 & \phi_a & -\phi_a & 0 \end{bmatrix} \tag{39}$$

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

$$r := 4$$
$$d := 1 \tag{40}$$

> $Estpar(D1, pars, 0);$

$$\left\{ f\left(\phi_1, \phi_a, im, \rho, p\right) = \_F1\left(\phi_a, \rho, \frac{\phi_1\,\rho}{2} + im, p\right) \right\} \tag{41}$$

> # All 4 data sets:

> kappa := $convert(\langle kappa1, kappa2s, kappa3s, kappa4\rangle, Vector);$

$$\kappa := \begin{bmatrix} -\left(x_{0,1}+x_{0,2}\right)\left(\phi_1\,\rho+2\,im+2\,\phi_a\right) \\ \phi_1\,\rho+2\,im+2\,\phi_a \\ \phi_a\,p \\ \phi_a\,(1-p) \\ \phi_1\,p \\ \phi_a\,p \\ \phi_1\,(1-p) \\ \rho\,R_t \end{bmatrix} \tag{42}$$

> $pars := \langle$phi[1], phi[a], im, rho, $p\rangle$:

> $D1 := Dmat($kappa, $pars)$

$$D1 := \begin{bmatrix} -\left(x_{0,1}+x_{0,2}\right)\rho & \rho & 0 & 0 & p & 0 & 1-p & 0 \\ -2\,x_{0,1}-2\,x_{0,2} & 2 & p & 1-p & 0 & p & 0 & 0 \\ -2\,x_{0,1}-2\,x_{0,2} & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\left(x_{0,1}+x_{0,2}\right)\phi_1 & \phi_1 & 0 & 0 & 0 & 0 & 0 & R_t \\ 0 & 0 & \phi_a & -\phi_a & \phi_1 & \phi_a & -\phi_1 & 0 \end{bmatrix} \tag{43}$$

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

$$r := 5$$

(44)

$$d := 0 \tag{44}$$

>

>