```
// T53mixExpCat.sce
// MIXTURE ESTIMATION (descriptive categorical)
// - static normal componens
// - scalar y and multinomial discrete v
//   |v1 | 1 1 1 1 2 2 2 2
//   |v2 | 1 1 2 2 1 1 2 2
//   |v3 | 1 2 1 2 1 2 1 2
//   |y=1| . . . . . . . .
//   |y=2| . . . . . . . .
// Experiments
// - change simulated parameters pS
// - change initial pE (through their statistics)
// ---------------------------------------------------------------------
exec("ScIntro.sce",-1),
getd(), mode(0)

nd=1000;                                                              // 1
// PARMETERS                                                          // 2
pS=list();                                                            // 3
pS(1)=genThs(2,8,.5);                  // simulated regr. coefficants  // 4
pS(2)=genThs(2,8,.5);                                                 // 5
v(1,:)=genCateg(1,nd,[.4 .6]);                                        // 6
v(2,:)=genCateg(1,nd,[.7 .3]);                                        // 7
```

```
v(3,:)=genCateg(1,nd,[.1 .9]);                                          // 8
nc=length(pS);                                                          // 9
                                                                        // 10
// SIMULATION                                                           // 11
for t=1:nd                                                              // 12
  jS=ceil(2*randu());                  // pointer value generation     // 13
  cS(t)=jS;                                                             // 14
  y(t)=genCategXY(v(:,t),pS(jS),[2 2 2]); // output value generation    // 15
end                                                                     // 16
                                                                        // 17
// INITIALIZATION                                                       // 18
S=list(); pE=list();                                                    // 19
ka=[1 1];                              // initial counter               // 20
for j=1:nc                                                              // 21
  S(j)=randu(2,8);                     // initial statistics            // 22
  pE(j)=fnorm(S(j),1);                 // initial parameters            // 23
end                                                                     // 24
                                                                        // 25
// TIME LOOP                                                            // 26
yp=zeros(nd,1);                                                         // 27
for t=1:nd                                                              // 28
  // estimation                                                         // 29
  for j=1:nc                                                            // 30
```

```
    z=xt2col(v(:,t),[2 2 2]);          // coding of variables v       // 31
    q(j)=pE(j)(y(t),z);                // proximity                   // 32
  end                                                                 // 33
  w=q/sum(q);                          // weights                     // 34
  wt(:,t)=w;                                                          // 35
  z=xt2col(v(:,t),[2 2 2]);            // coding of variables v       // 36
  for j=1:nc                                                          // 37
    S(j)(y(t),z)=S(j)(y(t),z)+w(j);    // update of statistics        // 38
    pE(j)=fnorm(S(j),1);               // parameter point estimates   // 39
  end                                                                 // 40
  // zero-step prediction                                             // 41
  z=xt2col(v(:,t),[2 2 2]);            // coding of variables v       // 42
  wp=dDel(w);                                                         // 43
  for j=1:nc                                                          // 44
    [nill,ypp]=max(pE(j)(:,z));        // output prediction           // 45
    yp(t)=yp(t)+wp(j)*ypp;                                            // 46
  end                                                                 // 47
end                                                                   // 48
                                                                      // 49
// RESULTS                                                          // 50
disp 'Simulated parameter values'                                     // 51
disp(pS(:))                                                           // 52
disp 'Final estimated parameters'                                     // 53
```

```
disp(pE(:))                                                          // 54
                                                                     // 55
s=(nd-100+1):nd;                                                     // 56
set(scf(2),'position',[900 100 800 400])  // plot data and prediction  // 57
plot(s,y(s),':ob',s,yp(s),':xr','markersize',8)                      // 58
title 'Data (b) and prediction (r)'                                  // 59
                                                                     // 60
[nill,cp]=max(wt,'r');                    // accuracy of classification // 61
disp 'Accuracy of classification'                                    // 62
ACCc=acc(cS,cp)                                                      // 63
                                                                     // 64
disp 'Accuracy of prediction'             // accuracy of prediction    // 65
ACCy=acc(y(:),yp(:))                                                 // 66
```

Description of the program

- Rows 3–10 define parameters for simulation.
- Rows 11–16 perform simulation with switching components. Simulated model is multivariate.
- Rows 19–24 prepare initial statistics and parameters for the estimation.
- Rows 27–48 represent the time loop.
  - Rows 30–33 compute logarithmic proximities.

- Rows 34–35 perform normalization of proximities and take their exponent.
- Rows 38–39 do statistics update and construction of point estimates of the parameters.
- Rows 42–47 construct the output zero-step prediction (either point one - expectation or generated one - expectation + noise).