```
// T41mixBasic.sce
// MIXTURE ESTIMATION (basic experiment)
// - static normal componens
// - scalar variables
// Experiments
// - change simulated expectations thS
// - change initial expectations thE (through the statistics S, ka)
// ----------------------------------------------------------------------
exec("ScIntro.sce",-1),
getd(), mode(0)

nd=100;                                                                    // 1
// SIMULATION                                                              // 2
thS=[1 6 10];                     // simulated expectations                // 3
for t=1:nd                                                                 // 4
  c(t)=ceil(3*randu());           // pointer                               // 5
  y(t)=thS(c(t))+randn();         // output                                // 6
end                                                                        // 7
// ESTIMATION                                                              // 8
// initialization                                                          // 9
S=[5 10 15];                      // initial sum statistics                // 10
ka=[1 1 1];                       // initial counter                       // 11
thE=S./ka;                        //initial point estimates                // 12
```

```
nc=length(S);
// time loop
for t=1:nd
  for j=1:nc
    q(j)=GaussN(y(t),thE(j),1);    // proximity
  end
  w=q/sum(q);                      // weights
  for j=1:nc
    S(j)=S(j)+w(j)*y(t);           // update of sum statistics
    ka(j)=ka(j)+w(j);              // update of counter
    thE(j)=S(j)/ka(j);             // point estimates
  end
  tht(:,t)=thE';                   // remember for plot
end

// RESULTS
set(scf(),'position',[800 100 600 400])
plot(tht')
title 'Evolution of estimated parametrs'
disp 'Simulated parameter values'
disp(thS)
disp 'Final estimated parameters'
disp(thE)
```

Description of the program

This is the basic mixture estimation program which shows the most concise structure of the problem. To be specific, we use the very simple scalar static normal model whose parameter is is the expectation, the statistics are sum S and counter κ and the point estimate is the average S/κ. The model can be substituted by other suitable one (from the exponential family) with its statistics and point estimates. The selection of the initial statistics is not solved here. However, it is very important and it will be demonstrated later.

- Row 3 defines the component parameters for simulation
- Rows 4–7 perform simulation.
  - First the value of the pointer c_t (the active component) is determined and
  - then the output y_t is generated from the active component.
- Rows 10–12 specify the initialization. It is accomplished through the statistics so that performing the point estimates of parameters we obtain the desired values.
- Row 13 derives the number of components from the length of the statistics.
- Rows 15–26 perform the time loop for estimation.
  - Rows 16–18 compute the proximities of the measured data y_t to individual components.
  - Row 19 constructs weights by normalization of the proximities
  - Rows 20–23 perform the weighted update of statistics S and κ for all components.
  - Row 24 constructs the point estimates of the component parameters.