State-space model

Anders Nielsen & Arni Magnusson

The Model

- Underlying stock size at time i is called X_i, and is expected to follow a random walk:

$$X_i = X_{i-1} + \eta_i \ , \ \text{where } \eta_i \sim N(0, \sigma_\eta^2)$$

- Our observations are:

$$Y_i = X_i + \varepsilon_i \ , \ \text{where } \varepsilon_i \sim N(0, \sigma_Y^2)$$

and

$$Z_i = X_i + \tau_i \ , \ \text{where } \tau_i \sim N(0, \sigma_Z^2)$$

- The two variables are not always observed. Missing values are coded as '-1000'

The data file

```
# N
 100
#T Y Z
1 10.49 9.94
2 11 10.51
3 9.72 10.77
4 10.52 10.91
5 12.89 14.13
6 13.77 14.08
7 14.18 13.77
8 17.19 17.64
9 17.98 18.57
10 18.44 18.95
11 16.59 15.41
12 15.92 16.21
13 15.73 15.75
14 16.75 17.02
15 17.89 18.68
16 17.89 17.88
17 16.94 17.7

-- cut ---
```

The AD Model Builder file

```
DATA_SECTION
  init_int N
  init_matrix tyz(1,N,1,3)
  vector t(1,N)
  vector y(1,N)
  vector z(1,N)
  !! t=column(tyz,1);
  !! y=column(tyz,2);
  !! z=column(tyz,3);

PARAMETER_SECTION
  init_number logSdx
  init_number logSdy
  init_number logSdz
  random_effects_vector x(1,N);
  objective_function_value jnll;

PROCEDURE_SECTION
  jnll=0.0;
  for(int i=2; i<=N; ++i){
    step(x(i-1),x(i),logSdx);
  }
  for(int i=1; i<=N; ++i){
    obs(x(i),logSdy,logSdz,i);
  }
```

```
SEPARABLE_FUNCTION void step(const dvariable& x1, const dvariable& x2, const dvariable& logSdx)
  dvariable var=exp(2.0*logSdx);
  jnll+=0.5*(log(2.0*M_PI*var)+square(x2-x1)/var);

SEPARABLE_FUNCTION void obs(const dvariable& x, const dvariable& logSdy, const dvariable& logSdz, int
  dvariable var=exp(2.0*logSdy);
  if(y(i)>-999){
    jnll+=0.5*(log(2.0*M_PI*var)+square(x-y(i))/var);
  }
  var=exp(2.0*logSdz);
  if(z(i)>-999){
    jnll+=0.5*(log(2.0*M_PI*var)+square(x-z(i))/var);
  }

  //TOP_OF_MAIN_SECTION
  //gradient_structure::set_MAX_NVAR_OFFSET(3000);
```

The result