LSD Pharmacokinetic Model

Kurt Bryan and SIMIODE

A worksheet to start the analysis of the two-compartment pharmacokinetic LSD model in Section 6.5.1.

The Data: Sampling times, in hours

```
times = [1/12, 1/4, 1/2, 1, 2, 4, 8]
```

The concentration data (mg/liter) for each subject at the seven times above is shown below (with "0" for the missing data point at time t = 4 hours for subject 3).

```
concdata = [11.1, 7.4, 6.3, 6.9, 5, 3.1, 0.8; 10.6, 7.6, 7, 4.8, 2.8, 2.5, 2;8.7, 6.7, 5.9, 4.3
    6.4, 6.3, 5.1, 4.3, 3.4, 1.9, 0.7]
```

A plot of the subject concentration data.

```
scatter(times,concdata(1,:))
hold on
scatter(times,concdata(2,:))
scatter(times,concdata(3,:))
scatter(times,concdata(4,:))
scatter(times,concdata(5,:))
legend('Subject 1','Subject 2','Subject 3','Subject 4','Subect 5')
hold off
```

We also have VP = 0.163*M liters and VT = 0.115*M where M is the subject mass in kg, and the initial dose is 2 mg per kg of body mass, so the actual initial dose in mg is 2*M. That means initial concentration of LSD in the blood is 2*M/(0.163*M) = 12.27 mg/liter for each subject (in theory).

With cP(t) as the plasma concentration and cT(t) as the tissue concentration we have cP(0) = 12.27 foreach subject and cT(0) = 0.

The ODE Model: The model for cP(t) and cT(t) of Section 6.5.1 (equation (6.67)) is

```
syms cP(t) cT(t) ka kb ke;
deP = 0.163*diff(cP(t),t) == (-0.163*kb - 0.163*ke)*cP(t) + 0.115*ka*cT(t)
deT = 0.115*diff(cT(t),t) == 0.163*kb*cP(t) - 0.115*ka*cT(t)
```

where ka, kb, and ke are the constants to be determined from the concentration data.

We can have Matlab solve these ODEs symbolically with initial data cP(0) = 12.27 and cT(0) = 0. The resulting solution is quite complicated, so the display is suppressed by putting a semicolon after dsolve and the assignment statements (remove it if you want to see the solution.)

```
sol = dsolve([deP, deT],[cP(0)==12.27,cT(0)==0]);
cPsol(t) = sol.cP;
cTsol(t) = sol.cT;
```

Fitting the Data: Construct the least squares functional that depends on ka, kb, and ke. Omit missing data point for subject 3 at time 4 hours.

```
syms SS(ka,kb,ke);
SS(ka,kb,ke) = 0;
for i=1:5
    SS(ka,kb,ke) = SS(ka,kb,ke) + sum((cPsol(times)-concdata(i,:)).^2);
end
SS(ka,kb,ke) = SS(ka,kb,ke) - (cPsol(times(6))-concdata(3,6)).^2; %Remove missing data point's
```

Now we need to minimize SS as a function of ka, kb, ke. This can be done by setting partial derivatives equal to zero, or by using Matlab's ***fminsearch*** algorithm to seek the minimizer, after converting "SS" to a Matlab function "SSm":

```
SSm = matlabFunction(SS,'Vars',{[ka kb ke]});
```

See the script "paramest_demo.mlx" for ***fminsearch***.