```
function dy = Morris_Lecar_2(t,y,I)
%SNLC
%Iapp
Iapp_SNLC = I;

%output Variables (Anthony)
VSNLC=y(1);
nSNLC=y(2);

%define Global Variables (Anthony)
Cm = 20 ; %microfarad/cm^2
ECa=120; %millivolts
gK=8;% millisiemens/ cm^2
EK=-84; %millivolts
gL=2;% millisiemens/ cm^2
EL=-60;%millivolts
v1=-1.2; %millivolts
v2= 18 ; %millivolts

%_____________________SNLC functions_____________________
%additional Parameters for (SNLC) Bifurcation Class 1 Neurons (Eric)
v3_SNLC = 12; % in mV
v4_SNLC = 17.4; % in mV
phi_SNLC = 0.067; % in ms^-1
gCa_SNLC=4.4; % millisiemens/ cm^2

%Other supporting functions in functions of voltage.
m_inf_SNLC = (0.5*(1+tanh((VSNLC-v1)/v2))); %(Anthony)
n_inf_SNLC = (0.5*(1+tanh((VSNLC-v3_SNLC)/v4_SNLC))); %(Anthony)
tau_n_SNLC = (1/(cosh((VSNLC-v3_SNLC)/(2*v4_SNLC)))); % (Eric)

%differential equations for hopf
% dn/dt(Anthony)
dnSNLC_over_dt = phi_SNLC*(n_inf_SNLC - nSNLC) / tau_n_SNLC;
% dV/dt(Eric)
P1_SNLC = gL*(VSNLC - EL); %leak component
P2_SNLC = gK * nSNLC * (VSNLC - EK); % pottasium component
P3_SNLC = gCa_SNLC * m_inf_SNLC * (VSNLC-ECa); %Calcium component
dVSNLC_over_dt = (Iapp_SNLC - P1_SNLC - P2_SNLC - P3_SNLC) / Cm; %combine
%_____________________________________________________________
%Wrapper function (Eric)
dy = [dVSNLC_over_dt;dnSNLC_over_dt];

end
```

```
Not enough input arguments.

Error in Morris_Lecar_2 (line 5)
Iapp_SNLC = I;
```

Published with MATLAB® R2023a