Narxnet

theta = dane\_NN\_test2.signals(1).values;

time = dane\_NN\_test2.time;

figure

plot(time, theta)

xlabel('Time')

ylabel('Theta')

grid on

y\_begin\_sample = 702;

y\_end\_sample = 4802;

training\_data\_input\_1 = theta(y\_begin\_sample-2:y\_end\_sample-2)

training\_data\_input\_2 = theta(y\_begin\_sample-1:y\_end\_sample-1)

training\_data\_output = theta(y\_begin\_sample:y\_end\_sample)

time\_train = (time(y\_begin\_sample:y\_end\_sample) - time(y\_begin\_sample))'

%wyjscie jest wejściem

x\_train = training\_data\_output'

inputSeries = num2cell(x\_train)

targetSeries = num2cell(x\_train)

net = narxnet(1:2, 1, 2);

[Xs,Xi,Ai,Ts] = preparets(net,targetSeries,{},inputSeries)

net = train(net,Xs,Ts,Xi,Ai);

[y\_net,Xf,Af] = net(Xs,Xi,Ai);

y\_net = cell2mat(y\_net)

figure();

hold on

plot(training\_data\_output)

plot(y\_net)

hold off

legend("Dane treningowe", 'Model sieci neuronowej')

xlabel('Time [s]')

ylabel('Theta [rad]')

title('Porownanie danych treningowych')

grid on

% err = immse(y\_train, y\_net)

% err = y\_net-y\_train

% err\_degree = err \* 180/pi

% figure;

% plot(err)

figure;

plot(err\_degree)