%-----GENERAZIONE INPUT E OUTPUT AGLI ISTANTI PRECEDENTI

if ritardi\_output ~=0

j=1+ (ritardi\_output)- (ritardi\_input)

else

j=1

end

input\_train=[] %inizializzo vettore di appoggio

input\_test=[]

for i= ritardi\_input:-1:0 %for(i=ritardi\_input; i==0 ; i-- )

input\_train= [inputl(j:end-i,1) input\_train];

input\_test= [inputt(j:end-i,1) input\_test];

j=j+1

end

output\_train=[]

output\_test=[]

if ritardi\_output ~=0

j=1

else

j= 1+ritardi\_input;

end

for i= ritardi\_output:-1:0 %for(i=ritardi\_output; i==0 ; i-- )

output\_train= [outputl(j:end-i,1) output\_train];

output\_test= [outputt(j:end-i,1) output\_test];

j=j+1

end

input\_train = [input\_train output\_train(:,2:end)];

input\_test = [input\_test output\_test(:,2:end)];

%-------GENERAZIONE RETE E LEARNING

Rete = newff(input\_train',(output\_train(:,1))',neuroni); %creo nuova rete

Rete.trainParam.show= show; %imposto parametro show

Rete.trainParam.lr=lr; %imposto parametro lr

Rete.trainParam.epochs=epochs; %imposto parametro epochs

Rete = train(Rete,input\_train',(output\_train(:,1))'); %learning rete

output\_rete= Rete(input\_train') %genero output dalla rete post learning

assignin('base','output\_rete',output\_rete)

performancel =perform(Rete, output\_rete, (output\_train(:,1))')

set(handles.edit17,'String',num2str(performancel))

output\_rete\_t= Rete(input\_test') %genero output dalla rete post leatning passandogli gli input di test

assignin('base','output\_rete\_t',output\_rete\_t)

performancet =perform(Rete, output\_rete\_t, (output\_test(:,1))')

set(handles.edit18,'String',num2str(performancet))

set(handles.edit20,'String',avviso)

%-------- ERRORI

errore\_l = output\_rete - (output\_train(:,1)')

assignin('base','errore\_l',errore\_l)

errore\_t = output\_rete\_t - (output\_test(:,1)')

assignin('base','errore\_t',errore\_t)