clear all

clc

format short g

[tumData] = xlsread('D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\DataSet.xlsx','Sayfa1','D7:H62')

tumverip=tumData(1:56,1:4) %girdi seti

tumverit=tumData(1:56,5) %çıktı seti

ptr=tumverip(1:38,1:4) %ptr eğitim girdisi

ttr=tumverit(1:38) %ttr eğitim çıktısı

val.P=tumverip(39:47,1:4) %val.P doğrulama girdisi

val.T=tumverit(39:47) %val.T doğrulama çıktısı

test.P=tumverip(48:56,1:4) %test.P test girdisi

test.T=tumverit(48:56) %test.T test çıktısı

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

[tumveripn,ps]=mapminmax(tumverip') %transpoze ve normalize işlemi ps girdi seti normalize bilgisi

[tumveritn,ts]=mapminmax(tumverit') %transpoze ve normalize işlemi ts çıktı seti normalize bilgisi

%[tumveripn,minp,maxp,tumveritn,mint,maxt]=premnmx(tumverip',tumverit') %Transpoze İşlemi

tumveripn=tumveripn'

tumveritn=tumveritn'

ptrn=tumveripn(1:38,1:4) %ptrn normalize eğitim girdisi..

ttrn=tumveritn(1:38) %ttrn normalize eğitim çıktısı

valn.P=tumveripn(39:47,1:4) %valn.P normalize doğrulama girdisi..

valn.T=tumveritn(39:47) %valn.T normalize doğrulama çıktısı

testn.P=tumveripn(48:56,1:4) %testn.P normalize test girdisi..

testn.T=tumveritn(48:56) %testn.T normalize test çıktısı

ptrn=ptrn'

ttrn=ttrn'

valn.P=valn.P'

valn.T=valn.T'

testn.P=testn.P'

testn.T=testn.T'

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

net=newff(minmax(ptrn),[3 4 1], {'tansig','tansig','purelin'},'trainlm')

net.trainParam.epochs=500

net.trainParam.goal=0.0001

net.trainParam.lr=0.4

net.trainParam.mc=0.6

net.trainParam.max\_fail=30

[net,tr]=train(net,ptrn,ttrn,[],[],valn,testn)

an=sim(net,ptrn) %normalize eğitim girdisi için ağı simüle et

avn=sim(net,valn.P) %normalize doğrulama girdisi için ağı simüle et

atn=sim(net,testn.P) %normalize test girdisi için ağı simüle et

a=mapminmax('reverse',an,ts) %an yi ts bilgisi ile denormalize et %a=postmnmx(an,mint,maxt)

av=mapminmax('reverse',avn,ts) %avn yi ts bilgisi ile denormalize et %av=postmnmx(avn,mint,maxt)

at=mapminmax('reverse',atn,ts) %atn yi ts bilgisi ile denormalize et %at=postmnmx(atn,mint,maxt)

[a]=a'

[av]=av'

[at]=at'

[m,n]=size(ttr)

mape\_egitim=sum(abs((ttr-a)./ttr)\*100)/m

rmse\_egitim=sqrt(sum((ttr-a).^2)/m)

[m,n]=size(val.T)

mape\_val=sum(abs((val.T-av)./val.T)\*100)/m

rmse\_val=sqrt(sum((val.T-av).^2)/m)

[m,n]=size(test.T)

mape\_test=sum(abs((test.T-at)./test.T)\*100)/m

rmse\_test=sqrt(sum((test.T-at).^2)/m)

[mape\_egitim, mape\_val, mape\_test]

[rmse\_egitim, rmse\_val, rmse\_test]

figure(1)

[m,b,r]=postreg(a',ttr')

saveas(gcf,'D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\1.png','png')

saveas(gcf,'D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\1.fig','fig')

figure(2)

[m,b,r]=postreg(av',val.T')

saveas(gcf,'D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\2.png','png')

saveas(gcf,'D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\2.fig','fig')

figure(3)

[m,b,r]=postreg(at',test.T')

saveas(gcf,'D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\3.png','png')

saveas(gcf,'D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\3.fig','fig')

figure(4)

plot(ttr,'-or','LineWidth',1)

hold on

plot(a,'-.\*b','LineWidth',1)

title('Eğitim Verileri','FontSize',14)

legend('GERÇEK','YSA','Location','NorthEast')

xlabel('Örnekler','FontSize',14);

ylabel('Cutting Power','FontSize',14)

saveas(gcf,'D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\4.png','png')

saveas(gcf,'D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\4.fig','fig')

figure(5)

plot(val.T,'-or','LineWidth',1)

hold on

plot(av,'-.\*b','LineWidth',1)

title('Doğrulama Verileri','FontSize',14)

legend('GERÇEK','YSA','Location','NorthEast')

xlabel('Örnekler','FontSize',14)

ylabel('Cutting Power','FontSize',14)

saveas(gcf,'D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\5.png','png')

saveas(gcf,'D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\5.fig','fig')

figure(6)

plot(test.T,'-or','LineWidth',1)

hold on

plot(at,'-.\*b','LineWidth',1)

title('Test Verileri','FontSize',14)

legend('GERÇEK','YSA','Location','NorthEast')

xlabel('Örnekler','FontSize',14)

ylabel('Cutting Power','FontSize',14)

saveas(gcf,'D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\6.png','png')

saveas(gcf,'D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\6.fig','fig')

save ('D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\MyApp')

olcutler=[mape\_egitim mape\_val mape\_test rmse\_egitim rmse\_val rmse\_test];

xlswrite('D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\Ölçütler.xlsx',olcutler,'Sayfa1','G2');

alldata=[[ptr,ttr,a,ttr-a,(((ttr-a).\*100)./ttr)];[val.P,val.T,av,(val.T-av),(((val.T-av).\*100)./val.T)];[test.P,test.T,at,(test.T-at),(((test.T-at).\*100)./test.T)]];

xlswrite('D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\Ölçütler.xlsx',alldata,'Sayfa3','G2');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Burdan Sonrası Ayrı Bir İşlem Ağ Kaydedildikten sonra ilgili veriler olcutdata değişkeni ile çıkarılır..

load MyApp

%MyApp isimli mat dosyasını açar...!Current Folderda Açık olmalı...!

%%olcutdata=[(((ttr-a).\*100)./ttr)(((val.T-av).\*100)./val.T)(((test.T-at).\*100)./test.T)];

%olcutdata=[[ptr,ttr,a,ttr-a,(((ttr-a).\*100)./ttr)];[val.P,val.T,av,(val.T-av),(((val.T-%av).\*100)./val.T)];[test.P,test.T,at,(test.T-at),(((test.T-at).\*100)./test.T)]];

%xlswrite('D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\Ölçütler.xlsx',olcutdata,'Sayfa4','G2');

%plotperform(net) %kurulan ağın mse ve iterasyon sayısını gösterir...!

%view(net) kurulan ağ yapısını gösterir

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

[depo]= xlsread('D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\DataSet.xlsx','Sayfa1','D63:G66')

pnew=depo % pnew=depo;

[pnewn,ps]=mapminmax(pnew') % pnewn=tramnmx(pnew',minp,maxp);

anewn=sim(net,pnewn) % anewn=sim(net,pnewn);

anew=mapminmax('reverse',anewn,ts) % anew=postmnmx(anewn,mint,maxt);

[pnew anew'] % [pnew anew']

xlswrite('D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\optimum.xlsx',[pnew,anew'],'Sayfa5','E8');

figure(7)

plot(tr.epoch, tr.perf, 'LineWidth' , 1)

saveas(gcf,'D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\7.png','png')

saveas(gcf,'D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\7.fig','fig')

net.IW{1,1}

net.b{1}

net.LW{2,1}

net.b{2}

net.LW{3,2}

net.b{3}

xlswrite('D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\sonuc.xlsx', net.IW{1,1}, 'Sayfa3', 'C3');

xlswrite('D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\sonuc.xlsx', net.b{1}, 'Sayfa3', 'M3');

xlswrite('D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\sonuc.xlsx', net.LW{2,1}, 'Sayfa3', 'C14');

xlswrite('D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\sonuc.xlsx', net.b{2}, 'Sayfa3', 'M14');

xlswrite('D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\sonuc.xlsx', net.LW{3,2}, 'Sayfa3', 'C25');

xlswrite('D:\DERS İÇERİKLERİ\YAPAY SİNİR AĞLARI\ArtificialNeuralNetworks2\sonuc.xlsx', net.b{3}, 'Sayfa3', 'M25');