



BENEMERITA UNIVERSIDAD AUTONOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN

LIC. CIENCIAS DE LA COMPUTACIÓN

SEMESTREAL - OTOÑO

INTELIGENCIA ARTIFICIAL

ALUMNO: ANDRÉS CRUZ CHIPOL MATRICULA: 201817637

DOCENTE: LUIS RENE MARCIAL CASTILLO

PRACTICA DE LABORATORIO

INDICACIONES

<http://archive.ics.uci.edu/ml/datasets/Skin+Segmentation> (Para Skin_NonSkin)

1. Ir a DataFolder
2. Bajar Skin_NonSkin.txt

<http://archive.ics.uci.edu/ml/datasets/Fertility> (Fertility)

1. Ir a DataFolder
2. Bajar fertility_Diagnossis.txt

<http://archive.ics.uci.edu/ml/datasets/Dry+Bean+Dataset> (Frijol seco)

1. Ir a DataFolder
2. Bajar DryBeanDataset.zip En excel vienen los datos

<http://archive.ics.uci.edu/ml/datasets/Diabetic+Retinopathy+Debrecen+Data+Set>

Bajar messidor_features.arff

Divida las bases de datos en dos partes A y B (ambas con la mitad de cada una de las clasificaciones) Una la debe usar para entrenar la red neuronal y la otra para ver el porcentaje de éxito en los problemas no vistos.

Escribir un reporte para la red multicapa (newff), Estructura de red neuronal, Cantidad de capas, neuronas por capas, funciones de activación, métodos de entrenamiento)

Reporte la cantidad de clasificación correcta en la base de entrenamiento y en la fase de prueba.

Probar con la red neuronal en base radial y red neuronal probabilista, reportando sus resultados para cada base de datos.

Nombre del archivo: IA Reporte RN ApellidoPaterno ApellidoMaterno.pdf

Diabetes

```
diabetesA.txt X
C:\Users\andy_ > Desktop > matlab > Diabetes > diabetesA.txt
553 1,1,26,26,26,25,23,12,66,86373,31.334414,8.615861,2.649828,0.603659,0.37639
554 1,0,69,67,62,54,49,36,12,144417,4.758572,0.373131,0.053743,0,0,0,0.527866
555 1,1,71,69,66,62,58,43,41,585991,13.626855,4.13593,0.003274,0.006168,0,0,0
556 1,1,22,21,20,19,15,12,59,628898,20.975956,11.511156,1.105663,0.142101,0.011
557 1,0,91,90,86,76,68,48,6,348365,0.882015,0.269837,0,0,0,0,0.495467,0.12381
558 1,1,19,18,18,15,13,10,74,284722,43.47836,0.782888,0.271217,0.003847,0.00384
559 1,1,38,38,38,35,33,26,14,20862,5.887533,0.370426,0.003139,0,0,0,0.518991,
560 1,1,19,17,16,15,10,6,27,413234,9.422,0.773448,0.027729,0,0,0,0.539483,0
561 1,1,19,19,17,17,14,9,97,439931,44.077855,15.029826,1.765179,0.135227,0.0051
562 1,0,63,61,59,56,50,29,34,052857,14.377599,5.018964,2.007278,0.178699,0.05003
563 1,1,19,19,18,16,14,8,67,000639,13.783533,5.318499,1.299765,0.185681,0.02589
564 1,1,5,4,4,4,3,3,81,52601,21.644911,9.326421,1.330601,0.240261,0.014253,0,0
565 1,1,42,38,32,24,15,9,69,50377,23.654779,12.949117,5.937161,3.242952,1.84633
566 1,1,58,56,51,44,35,30,3,557701,1.684824,0.539694,0.298131,0.151359,0,0,0,0
567 1,0,19,19,17,14,8,8,137,038933,15.034527,4.659757,0.567083,0.184118,0.09363
568 1,1,41,41,40,37,35,28,7,486693,2.386675,0.157655,0,0,0,0,0.518948,0.09365
569 1,1,36,35,32,29,27,21,3,203759,0.024433,0.001527,0,0,0,0,0.545851,0.10231
570 1,1,13,13,12,11,11,3,48,546862,27.189621,1.059144,0.075935,0,0,0,0.493149
571 1,1,51,43,42,36,29,21,29,584628,9.536418,1.358037,0.516803,0.143499,0.05303
572 1,1,70,68,64,63,58,38,66,751648,32.166708,14.077515,3.378666,0.053589,0.0214
573 1,1,72,69,68,62,56,46,24,046639,12.753548,2.537867,0.204864,0.053589,0.0214
574 1,1,22,22,19,17,15,11,5,468078,2.519937,0.155571,0,0,0,0,0.546808,0.17405
575 1,1,114,99,94,84,76,49,31,648833,13.188407,2.217076,0.173352,0.021289,0.013
```

```
diabetesB.txt X
C:\Users\andy_ > Desktop > matlab > Diabetes > diabetesB.txt
556 1,1,71,62,57,47,37,23,95,232668,28.937086,20.983416,13.241118,9.436416,3.51
557 1,1,100,89,81,71,62,49,15,296399,7.512279,3.999727,2.96887,2.45115,0.922427
558 1,0,23,22,20,18,13,8,131,077221,47.825963,24.719398,3.161361,0.044599,0.0072
559 1,1,78,75,75,70,65,43,15,259422,8.751638,4.265453,0.185719,0.03958,0.004567
560 1,1,25,24,24,23,18,14,83,135316,46.049413,11.415154,1.663975,0.066478,0.0096
561 1,1,23,21,18,13,7,4,35,347026,9.43159,1.661299,0.247955,0.191132,0.169436,0
562 1,1,21,21,21,20,20,16,5,121229,1.694872,0.280699,0,0,0,0,0.4765,0.073226,
563 1,1,42,39,36,29,21,13,96,385981,24.281414,7.722666,0.641116,0.242004,0.1005
564 1,1,37,36,33,24,18,10,37,686807,4.978201,1.180336,0.184896,0.024986,0.007996
565 1,1,46,46,46,43,42,34,26,839986,16.25448,5.512953,0.316836,0.109734,0,0,0,0
566 1,1,86,84,83,82,80,63,24,226598,4.989495,1.692755,0.165296,0.009183,0.003061
567 1,1,56,54,54,51,48,36,8,804178,1.918597,0.168133,0,0,0,0,0.529728,0.100565
568 1,1,77,71,69,61,39,25,159,984908,94.032005,42.230588,16.562221,8.562842,5.1
569 1,1,65,60,48,31,24,13,159,723711,43.983882,17.081478,1.644218,0.248323,0.087
570 1,1,31,30,30,29,28,21,23,736589,12.865507,1.036667,0.072721,0.015473,0,0,0,0
571 1,1,3,3,3,3,3,1,1,22,266376,6.890557,0.731205,0.063583,0.007179,0.001026,0,0
572 1,1,12,12,10,9,8,4,51,440209,26.643732,6.51126,0.312865,0,0,0,0.520232,0.
573 1,1,31,31,31,27,22,17,22,404714,13.005926,1.196279,0.100579,0,0,0,0.55572
574 1,1,53,53,52,51,51,41,8,63201,3.852405,0.383902,0.290335,0.16546,0.109266,0
575 1,1,12,12,10,9,7,6,243,066702,26.650463,13.732893,0.099581,0.002075,0.00103
576 1,1,39,36,29,23,13,7,40,525739,12.604947,4.740919,1.07757,0.563518,0.32686
```

diabetesA.txt, 575 Datos

diabetesB.txt 576 Datos

Se Clasifica en = {0,1}

Fertilidad

```
fertilidadA.txt X
C:\Users\andy_ > Desktop > matlab > Fertilidad > fertilidadA.txt
28 1,0,58,0,0,1,0,1,-1,0.19,1
29 1,0,61,1,0,1,0,1,-1,0.63,1
30 1,0,56,1,0,0,0,1,-1,0.44,1
31 1,0,64,0,0,0,0,1,-1,0.63,1
32 1,0,58,1,1,1,0,0,8,0.0.44,1
33 1,0,56,1,1,1,0,1,-1,0.63,1
34 -1,0.78,1,1,0,1,0.6,-1,0.38,1
35 -1,0.78,1,0,1,0,1,-1,0.25,1
36 -1,0.56,1,0,1,0,1,-1,0.63,1
37 -1,0.69,1,0,0,0,1,-1,0.31,1
38 -1,0.53,1,1,1,0,0.8,1,0.5,1
39 -1,0.56,1,1,0,0,0.8,1,0.5,1
40 -1,0.58,1,0,1,-1,0.8,1,0.5,1
41 -1,0.56,1,0,0,0,1,-1,0.44,1
42 -1,0.53,1,1,0,1,1,0,0.31,1
43 -1,0.53,1,0,0,1,1,0.0.44,1
44 -0.33,0.56,1,0,0,0,1,-1,0.63,1
45 -0.33,0.72,1,1,0,0,0.6,1,0.19,1
46 -0.33,0.64,1,1,1,0,0.8,-1,0.31,1
47 -0.33,0.75,1,1,1,0,0.6,-1,0.19,1
48 -0.33,0.67,1,0,1,0,0.8,-1,0.19,1
49 -0.33,0.53,1,1,0,1,1,-1,0.75,1
50 -0.33,0.53,1,1,0,0,0.8,0.0.5,1
```

```
fertilidadB.txt X
C:\Users\andy_ > Desktop > matlab > Fertilidad > fertilidadB.txt
27 -1,0.53,1,0,0,1,0.8,-1,0.63,1
28 -1,0.78,1,0,1,1,1,1,0.25,1
29 -1,0.75,1,0,1,1,0.6,0.56,1
30 -1,0.72,1,1,1,1,0.8,-1,0.19,1
31 -1,0.53,1,1,0,1,0.8,-1,0.38,1
32 -1,1,1,0,1,1,0.6,0.25,1
33 -0.33,0.92,1,1,0,1,1,-1,0.63,1
34 -1,0.81,1,1,1,1,0.8,0.0.19,1
35 -0.33,0.92,1,0,0,1,0.6,-1,0.19,1
36 -0.33,0.86,1,1,1,1,1,-1,0.25,1
37 -0.33,0.89,1,1,0,0,0.6,1,0.31,1
38 -0.33,0.75,1,1,1,0,0.6,1,0.25,1
39 -0.33,0.75,1,1,1,1,0.8,1,0.25,1
40 -0.33,0.83,1,1,1,0,1,-1,0.31,1
41 -0.33,0.81,1,1,1,0,1,1,0.38,1
42 -0.33,0.81,1,1,1,1,0.8,-1,0.38,1
43 0.33,0.78,1,0,0,0,1,1,0.06,1
44 0.33,0.75,1,1,0,0,0.8,-1,0.38,1
45 1,0.58,1,0,0,0,0.6,1,0.5,1
46 -1,0.67,1,0,0,0,1,-1,0.5,1
47 -1,0.61,1,0,0,0,0.8,0.0.5,1
48 -1,0.67,1,1,1,0,1,-1,0.31,1
49 -1,0.64,1,0,1,0,1,0.0.19,1
50 -1,0.69,0,1,1,0,0.6,-1,0.19,1
```

FertilidadA.txt, 50 Datos

FertilidadB.txt, 50 Datos

Se Clasifica en = {0,1}

Frijoles

```
frijolesA.txt X
C:\Users\andy\Desktop> matlab > frijoles > frijolesA.txt
6774 31815,674,0769999999999,248,14273696310335,164,6193726586511,1,507372631510
6775 31810,670,763,247,10085909149850,164,23885788070516,1,504565138043930,0,747
6776 31824,665,2610000000000,249,0265506197205,163,0475839443729,1,5273243834436
6777 31820,658,6709999999999,232,21735782766538,175,02570343274405,1,3267614600
6778 31834,649,5899999999999,235,24444521735356,172,7409357757406,1,361833801356
6779 31838,658,144,234,44043693915376,173,37937863952226,1,3521817806636924,0,67
6780 31845,679,0879999999999,246,94779561724004,164,36585407655133,1,5024275997
6781 31846,657,514,238,00013926567956,170,00415572710017,1,4053135245069501,0,70
6782 31850,655,786,245,39365047996117,165,78315404575164,1,408208007619277,0,70
6783 31851,662,1539999999999,236,69133151016860,171,7051500012050,1,378475435339
6784 31855,664,685,251,8856927525980,161,25127214954438,1,5620694918858817,0,70
6785 31864,664,41,255,17161105424947,159,31183180860916,1,6017196559294395,0,70
6786 31868,658,384,242,874173550903,167,9745424400938,1,441136079449104,0,72007
6787 31870,659,448,245,5347346302757,165,48114396207024,1,4837626133719446,0,73
6788 31874,665,4630000000000,250,31418985630702,162,36108356100468,1,5417129792
6789 31877,666,1529999999999,247,2493381219206,164,47530771010872,1,50326113727
6790 31877,673,251,250,38000635949064,162,29942614826712,1,5427841998956355,0,70
6791 31878,667,718,254,70432306075935,159,71047899083343,1,594787480446345,0,7
6792 31879,659,67,246,5910633541789,165,26096953450195,1,49213455719748,0,7421
6793 31882,665,7030000000000,250,24620613473627,162,502956389392,1,5390486365906
6794 31884,666,0,249,53127200136607,163,3086858099934,1,5271314497234652,0,7557
6795 31884,665,008,248,6468228674213,163,55457878131807,1,5202681064374783,0,75
6796 31890,660,655,250,4172999306156,162,52250245071204,1,5400161710195134,0,70
```

```
frijolesB.txt X
C:\Users\andy\Desktop> matlab > frijoles > frijolesB.txt
6795 41962,755,945,286,8703906326574,186,58893664707284,1,5374458731991372,0,750
6796 41964,756,887,287,1641019821347,186,27110809373522,1,54164596389059,0,7610
6797 41966,746,121,273,5086784154536,195,44915348486924,1,3993853313702243,0,699
6798 41979,757,238,272,57992017454717,196,5807060243304,1,3866056628202896,0,692
6799 41992,763,279,293,22412213026803,182,87952247904974,1,603373183369172,0,781
6800 41995,765,763,284,0731777922192,188,5919566100672,1,506284694672948,0,74783
6801 42008,759,454,208,3327165874974,191,21813607804417,1,4660362366655286,0,731
6802 42008,773,1579999999999,294,4922025203752,181,84735861066202,1,619447237296
6803 42012,778,0430000000001,297,9128117073527,179,71183573961048,1,657725049000
6804 42023,746,715,272,9626857858725,196,3340473924524,1,390297248037917,0,69472
6805 42026,751,05,278,0711406529184,192,95676003429065,1,441105979409594,0,72006
6806 42042,771,515,288,08267411770544,186,34708959848243,1,545946731652371,0,762
6807 42047,768,9359999999999,292,97500733548736,183,1391409147197,1,59973998934
6808 42049,770,185,290,1634033084052,185,05168512513018,1,5680127587716883,0,774
6809 42070,763,489,289,0223733637728,186,12343389845492,1,5528532184800403,0,765
6810 42070,760,7009999999999,276,69165062642804,193,94536647332566,1,4266473886
6811 42097,759,6959999999999,288,72161204221896,185,94470544010682,1,55272833049
6812 42101,757,499,281,57639233416154,190,71313645035534,1,4764394187783645,0,73
6813 42139,759,321,281,53992791425856,191,18797890118992,1,4725817466785633,0,73
6814 42147,763,779,283,3826363799513,190,27573077099427,1,4893262279518744,0,741
6815 42159,772,237,295,1427409885261,182,20471589551335,1,619841393994313,0,7866
```

FrijolesA.txt, 6796 Datos

FrijolesB.txt, 6815 Datos

Se dividió la mitad de Cada tipo de Frijol

Clasificación = {0,1,2,3,4,5,6}

Skin

```
skinA.txt X
C:\Users\andy\Desktop> matlab > Skin > skinA.txt
122937 144,142,88,2
122938 144,142,88,2
122939 144,142,88,2
122940 144,142,88,2
122941 145,143,89,2
122942 145,143,89,2
122943 145,143,89,2
122944 145,143,89,2
122945 146,144,90,2
122946 146,144,90,2
122947 146,144,90,2
122948 146,144,90,2
122949 147,145,91,2
122950 147,145,91,2
122951 147,146,90,2
122952 151,150,92,2
122953 154,153,97,2
122954 154,153,97,2
122955 154,152,98,2
122956 156,154,100,2
122957 162,159,108,2
122958 167,164,113,2
122959 181,177,128,2
```

```
skinB.txt X
C:\Users\andy\Desktop> matlab > Skin > skinB.txt
122075 165,164,113,2
122076 165,164,113,2
122077 165,164,113,2
122078 165,164,113,2
122079 165,164,113,2
122080 164,163,112,2
122081 164,163,112,2
122082 164,163,112,2
122083 163,162,112,2
122084 163,162,112,2
122085 163,162,112,2
122086 163,162,112,2
122087 163,162,112,2
122088 163,162,112,2
122089 163,162,112,2
122090 163,162,112,2
122091 163,162,112,2
122092 163,162,112,2
122093 163,162,112,2
122094 163,162,112,2
122095 163,162,112,2
122096 163,162,112,2
122097 163,162,112,2
122098 255,255,255,2
```

skinA.txt, 122959 Datos

skinB.txt, 122098 Datos

Se dividió la mitad de Cada tipo de Frijol

Clasificación = {0,1}

Los Datos A Se usaron para Entrenamiento, Los Datos B para Probar.

[Fertilidad]

Red Neuronal Multicapa (Mejor Resultado Encontrado)

Cantidad de capas: 3

Neuronas por capa: 20 ,20,3

Funciones de activación: logsig, logsig, logsig

Métodos de entrenamiento: **CGP**

Código de Entrenamiento:

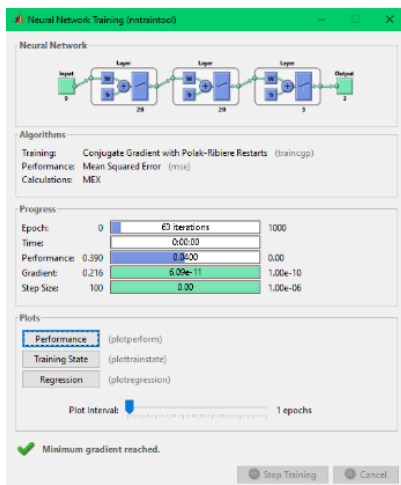
Porcentaje: 88%

```
function fertilidadRNN
load('Fertilidad\FertilidadA.txt')
[m,n] = size(fertilidadA);
X = fertilidadA(:,1:9);
t = [];
for i=1:m
    if(fertilidadA(i,10)==0) t=[t [1;0;0]];
    else if(fertilidadA(i,10)==1) t = [t;0;1;0];
    end
end
end
RNNunall = newff(minmax(X),[20,20,3],{'logsig','logsig','logsig'},'traincgp');
RNEntrenada.trainParam.epochs = 100000;
RNEntrenada.trainParam.min_grad = 1e-27;
RNEntrenada.trainParam.show = 1;
RNEntrenada.trainParam.time = 600;
RNEntrenada = train(RNNunall,X,t);
Y = sim(RNEntrenada, X)
tt = [];
for j=1:m
    [vm,pos]=max(Y(:,j));
    if(pos==1) tt=[tt [1;0;0]];
    else if(pos==2) tt = [tt [0;1;0]];
    end
end
end
%comparar Aciertos
aciertos = 0;
for i=1:m
    [a,pos1] = max(t(:,i));[aa,pos2] = max(tt(:,i));
    if (pos1 == pos2)
        aciertos = aciertos+1;
    end
end
porcentaje = (aciertos/m)*100
save fertiTrain RNEntrenada
save fertiTrainPorciento porcentaje
```

Código de Prueba:

Porcentaje: 88%

```
function fertilidadRNNPrueba
load('Fertilidad\FertilidadB.txt')
[m,n] = size(fertilidadB);
X = fertilidadB(:,1:9);
t = [];
for i=1:m
    if(fertilidadB(i,10)==0) t=[t [1;0;0]];
    else if(fertilidadB(i,10)==1) t = [t;0;1;0];
    end
end
end
load fertiTrain.mat
Y = sim(RNEntrenada, X);
tt = [];
for j=1:m
    [vm,pos]=max(Y(:,j));
    if(pos==1) tt=[tt [1;0;0]];
    else if(pos==2) tt = [tt [0;1;0]];
    end
end
end
%comparar Aciertos
aciertos = 0;
for i=1:m
    [a,pos1] = max(t(:,i));[aa,pos2] = max(tt(:,i));
    if (pos1 == pos2)
        aciertos = aciertos+1;
    end
end
porcentajePrueba = (aciertos/m)*100
save fertiPruebaPorciento porcentajePrueba
```



BGF Resultados con porcentajes muy bajos, LM en entrenamiento siempre nos da cerca de un 100% pero las pruebas tienen porcentajes bajos del 88%, CGF es parecido en porcentajes de entrenamiento, pero no de prueba, resulta un poco mas bajo que cgp, que solamente nos dio 1 vez el 90% en prueba, Todas las veces con distintas configuraciones resulto en un 88% como Máximo. Considero que tener más datos de entrenamiento, darían mejores resultados para este caso.

```
porcentaje =  
  
88  
  
>> fertilidadRNNPrueba  
  
porcentajePrueba =  
  
88
```

Red Neuronal Base Radial

Código de Entrenamiento

Porcentaje: 88%

```
MATLAB v
function fertilidadRadial
XA = load('Fertilidad\fertilidadA.txt');
[m,n] = size(XA);
X = XA(:,1:9)'; t=XA(:,10)';
RN = newrb(X,t,0.0,1.0,m,50);
Y = sim(RN,X)
ac = 0;
for i=1:m
    if round(Y(i)) == t(i)
        ac = ac+1;
    end
end
porcentaje = (ac/m)*100

save fertiRadialTrain RN
save fertiRadialPorcentaje porcentaje
```

Código de Prueba

Porcentaje: 88%

```
function fertilidadRadialPrueba
XA = load('Fertilidad\fertilidadB.txt');
[m,n] = size(XA);
X = XA(:,1:9)'; t=XA(:,10)';
load fertiRadialTrain.mat;
Y = sim(RN,X);
ac = 0;
for i=1:m
    if round(Y(i)) == t(i)
        ac = ac+1;
    end
end
porcentajePrueba = (ac/m)*100
save fertiRadialPruebaPorcentaje porcentajePrueba
```

Hola muy buenas a todos aquí andamos

```
>> fertilidadRadial
NEWRB, neurons = 0, MSE = 0.1056

porcentaje =

    88

>> fertilidadRadialPrueba

porcentajePrueba =

    88
```


Red Neuronal Probabilística

Código de Entrenamiento

Porcentaje: 100%

```
function fertProba
XA = load('Fertilidad\fertilidadA.txt');
[m,n] = size(XA);
X = XA(:,1:9)'; t=XA(:,10)';
t = t + 1;
tt = ind2vec(t);
RNP = newpnn(X,tt,0.001);
Ya = sim(RNP,X);
Y = vec2ind(Ya);
ac = 0;
for i=1:m
    if round(Y(i)) == t(i)
        ac = ac+1;
    end
end
porcentaje = (ac/m)*100
save fertiProTrain RNP
save fertiProTrainPorcentaje porcentaje
```

Código de Prueba

Porcentaje: 88%

MATLAB ▾

```
function fertProPrueba
XA = load('Fertilidad\fertilidadB.txt');
[m,n] = size(XA);
X = XA(:,1:9)'; t=XA(:,10)';
load fertiProTrain.mat
Ya = sim(RNP,X);
Y = vec2ind(Ya);
ac = 0;
for i=1:m
    if round(Y(i)) == t(i)
        ac = ac+1;
    end
end
porcentaje = (ac/m)*100

save fertiProTrainPorcentajePrueba porcentaje
```

```
>> fertilidadProbabilistico

porcentaje =

    100

>> fertilidadProbabilisticoPrueba

porcentaje =

    88
```

[Diabetes]

Red Neuronal Multicapa(Mejor Resultado Encontrado)

Trainlm

Cantidad de capas: 2

Neuronas por capa: 10,3

Funciones de activación: logsig, logsig

Métodos de entrenamiento: cgf

Código de Entrenamiento

Porcentaje: 82.4348%

```
function diabetesRNN
load('Diabetes\diabetesA.txt')

[m,n] = size(diabetesA);
X = diabetesA(:,1:19)';

t = [];
for i=1:m
    if(diabetesA(i,20)==0) t=[t [1;0;0]];
    else if(diabetesA(i,20)==1) t = [t,[0;1;0]];
    end
end
end
RNNUnAN = newff(minmax(X),[11,3],{'logsig','logsig'},'traingcf');
RNEntrenada.trainParam.epochs = 10000%;
RNEntrenada.trainParam.min_grad = 1e-27%;
RNEntrenada.trainParam.show = 1;
RNEntrenada.trainParam.time = 600;
RNEntrenada = train(RNNUnAN,X,t);
Y = sim(RNEntrenada, X)
tt = [];
for j=1:m
    [vm,pos]=max(Y(:,j));
    if(pos==1) tt=[tt [1;0;0]];
    else if(pos==2) tt = [tt [0;1;0]];
    end
end
end
%comparar Aciertos
aciertos = 0;
for i=1:m
    [a,pos1] = max(t(:,i));[aa,pos2] = max(tt(:,i));
    if (pos1 == pos2)
        aciertos = aciertos+1;
    end
end
end
porcentaje = (aciertos/m)*100
save diabetesTrain RNEntrenada
save diabetesTrainPorcentaje porcentaje
```

Código de Prueba

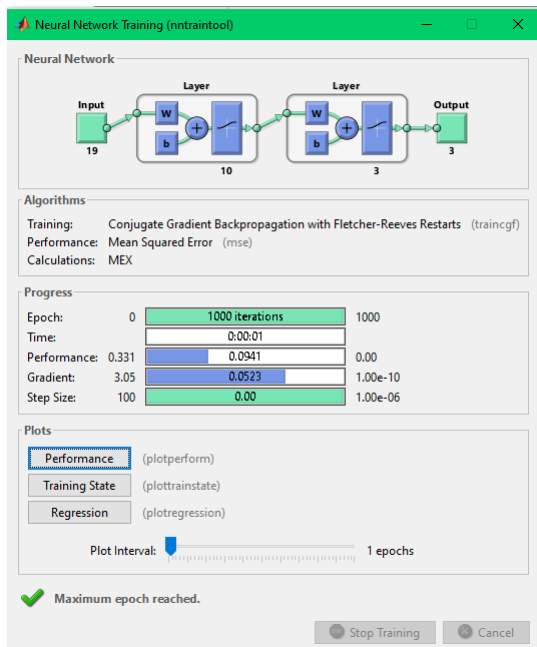
Porcentaje: 76.2153%

```
function diabetesPrueba
load('Diabetes\diabetesB.txt')

[m,n] = size(diabetesB);
X = diabetesB(:,1:19)';

t = [];
|
for i=1:m
    if(diabetesB(i,20)==0) t=[t [1;0;0]];
    else if(diabetesB(i,20)==1) t = [t,[0;1;0]];
    end
end
end
load diabetesTrain
Y = sim(RNEntrenada, X);

tt = [];
for j=1:m
    [vm,pos]=max(Y(:,j));
    if(pos==1) tt=[tt [1;0;0]];
    else if(pos==2) tt = [tt [0;1;0]];
    end
end
end
%comparar Aciertos
aciertos = 0;
for i=1:m
    [a,pos1] = max(t(:,i));[aa,pos2] = max(tt(:,i));
    if (pos1 == pos2)
        aciertos = aciertos+1;
    end
end
end
porcentaje = (aciertos/m)*100
```

Bfg da resultados cerca del 70% con distintos parámetros, es el peor método de entrenamiento para estos datos. Lm da resultados arriba del 92% en el entrenamiento, En datos de entrenamiento no pasa del 72%, Aumentando el numero de neuronas en cada capa el rendimiento en entrenamiento baja cerca de un 6% Y los datos de prueba Quedan alrededor de 72%, Esto con costo alto de tiempo. cgf obtiene en entrenamiento cerca del 74% y pruebas del 71%. Aumentando el numero de neuronas por capa incrementa el porcentaje en pruebas en

alrededor del 84%, En los datos de prueba un 76% El mejor resultado que hemos obtenido. Se aumento más el numero de capas y de neuronas por capa, no se obtuvieron mejores resultados, y en algunos csos, en promedio al 72%. cgf obtiene 96% En entrenamiento y 71.5% en pruebas.

Cfg con 2 capas 1000 y 3 neuronas obtiene 99.4783% en Datos de entrenamiento, con 70% con Datos de Pruebas.

```

porcentaje =

    82.4348

>> diabetesPruebaRNN

porcentaje =

    76.2153

```

Red Neuronal Base Radial

Código de Entrenamiento

Porcentaje: 100%

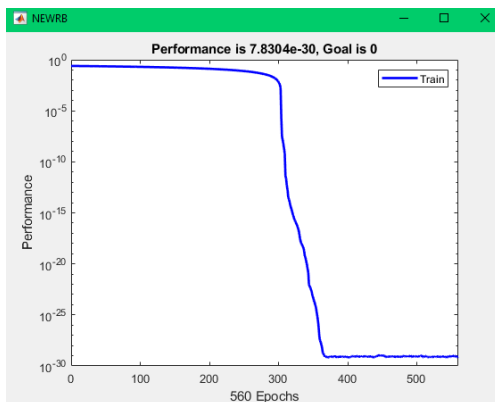
```
function diabetesRadial
XA = load('Diabetes\diabetesA.txt')
[m,n] = size(XA);
X = XA(:,1:19)'; t=XA(:,20)';
RN = newrb(X,t,0.0,1,m,20);
Y = sim(RN,X);
ac = 0;
for i=1:m
    if round(Y(i)) == t(i)
        ac = ac+1;
    end
end
porcentaje = (ac/m)*100

save diaRadialTrain RN
save diaRadialPorcentaje porcentaje
```

Código de Pruebas

Porcentaje: 53.1250%

```
function diabetesRadialPrueba
XA = load('Diabetes\diabetesB.txt')
[m,n] = size(XA);
X = XA(:,1:19)'; t=XA(:,20)';
load diaRadialTrain.mat
Y = sim(RN,X);
ac = 0;
for i=1:m
    if round(Y(i)) == t(i)
        ac = ac+1;
    end
end
porcentaje = (ac/m)*100
```



Encontrando los mejores parámetros para Base Radial, Con 100% en Entrenamiento. lo máximo que se logró encontrar de porcentaje en pruebas fue 53.1250%, Otros porcentajes con otros parámetros bajaban el porcentaje de las pruebas.

```
Command Window
NEWRB, neurons = 300, MSE = 0.00679474
NEWRB, neurons = 320, MSE = 1.12172e-15
NEWRB, neurons = 340, MSE = 1.84931e-20
NEWRB, neurons = 360, MSE = 3.01837e-28
NEWRB, neurons = 380, MSE = 7.56619e-30
NEWRB, neurons = 400, MSE = 7.00389e-30
NEWRB, neurons = 420, MSE = 8.16126e-30
NEWRB, neurons = 440, MSE = 8.69612e-30
NEWRB, neurons = 460, MSE = 8.08417e-30
NEWRB, neurons = 480, MSE = 9.60174e-30
NEWRB, neurons = 500, MSE = 8.12849e-30
NEWRB, neurons = 520, MSE = 7.64355e-30
NEWRB, neurons = 540, MSE = 8.592e-30
NEWRB, neurons = 560, MSE = 7.8304e-30

porcentaje =

    100
```

porcentajePrueba =

53.1250

Red Neuronal Probabilística

Código de Entrenamiento

Porcentaje: 73.5652%

```
function diaProbabilistica
XA = load('Diabetes\diabetesA.txt')
[m,n] = size(XA);
X = XA(:,1:19)'; t=XA(:,20)';
t = t + 1;
tt = ind2vec(t);
RN = newpnn(X,tt,17.8);
Ya = sim(RN,X);
Y = vec2ind(Ya);
ac = 0;
for i=1:m
    if round(Y(i)) == t(i)
        ac = ac+1;
    end
end
porcentaje = (ac/m)*100

save diabetesProTrain RN
save diabetesProTrainPorciernto porcentaje
```

porcentaje =

73.5652

Código de Prueba

Porcentaje: 66.4931%

```
function diabetesPruebaProba
XA = load('Diabetes\diabetesB.txt')
[m,n] = size(XA);
X = XA(:,1:19)'; t=XA(:,20)';
t = t + 1;
tt = ind2vec(t);
load diabetesProTrain.mat
Ya = sim(RN,X);
Y = vec2ind(Ya);
ac = 0;
for i=1:m
    if round(Y(i)) == t(i)
        ac = ac+1;
    end
end
porcentaje = (ac/m)*100

|
save diabPruebasPorciento porcentaje
```

porcentajePrueba =

66.4931

Se busco el mejor parámetro para la red, cambiando el ultimo paramero a 17.8, Teniendo otro diferente el porcentaje en la prueba bajaba, aun que en el entrenamiento subía no era la mejor opción para datos nuevos.

[Frijoles]

Red Neuronal Multicapa

Cantidad de capas: 5

Neuronas por capa: 30,26,18,17,7

Funciones de activación: logsig

Métodos de entrenamiento: Im

Código de Entrenamiento

```
function beansMl
load('Frijoles\frijolesA.txt')
[m,n] = size(frijolesA);
X = frijolesA(:,1:16)';
t = [];
for i=1:m
    if(frijolesA(i,17)==0) t=[t [1;0;0;0;0;0;0;0]];
    else if(frijolesA(i,17)==1) t = [t,[0;1;0;0;0;0;0;0]];
    else if(frijolesA(i,17)==2) t = [t,[0;0;1;0;0;0;0;0]];
        else if(frijolesA(i,17)==3) t = [t,[0;0;0;1;0;0;0;0]];
            else if(frijolesA(i,17)==4) t = [t,[0;0;0;0;1;0;0;0]];
                else if(frijolesA(i,17)==5) t = [t,[0;0;0;0;0;1;0;0]];
                    else if(frijolesA(i,17)==6) t = [t,[0;0;0;0;0;0;1;0]];
                        end
                    end
                end
            end
        end
    end
end
end
end
end
RMseall = newff(minmax(X),[30,26,10,17],{'logsig','logsig','logsig','logsig','logsig'},'trainlm');
RNEntrenada.trainParam.epochs = 10000k;
RNEntrenada.trainParam.min_grad = 1e-27k;
RNEntrenada.trainParam.show = 1;
RNEntrenada.trainParam.time = 600;
RNEntrenada = train(RNseall,X,t);
Y = sim(RNEntrenada, X);
tt = [];
for j=1:m
    [vm,pos]=max(Y(:,:));
    if(pos==1) tt=[tt [1;0;0;0;0;0;0;0]];
    else if(pos==2) tt = [tt [0;1;0;0;0;0;0;0]];
    else if(pos==3) tt = [tt [0;0;1;0;0;0;0;0]];
        else if(pos==4) tt = [tt [0;0;0;1;0;0;0;0]];
            else if(pos==5) tt = [tt [0;0;0;0;1;0;0;0]];
                else if(pos==6) tt = [tt [0;0;0;0;0;1;0;0]];
                    else if(pos==7) tt = [tt [0;0;0;0;0;0;1;0]];
                        else if(pos==8) tt = [tt [0;0;0;0;0;0;0;1]];
                            end
                        end
                    end
                end
            end
        end
    end
end
end
end

%comparar Aciertos
aciertos = 0;
for i=1:m
    [a,pos1] = max(t(:,i));[aa,pos2] = max(tt(:,i))
    if (pos1 == pos2)
        aciertos = aciertos+1;
    end
end

porcentaje = (aciertos/m)*100

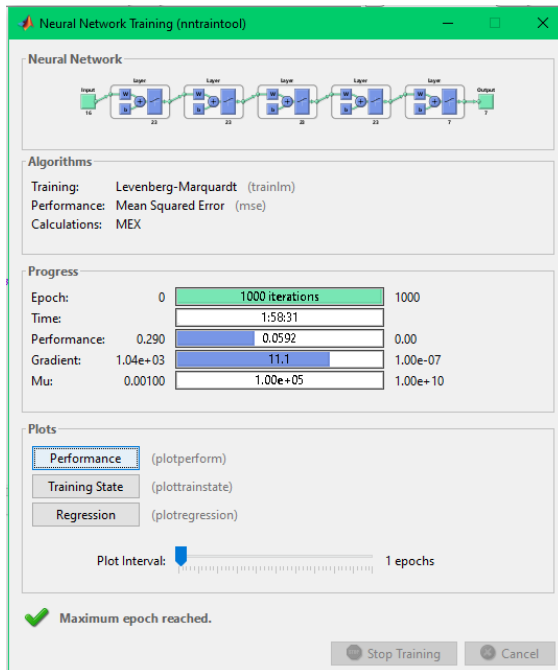
save beansTrain RNEntrenada
save diabetesTrainPorciernto porcentaje
```

Código de Pruebas

```

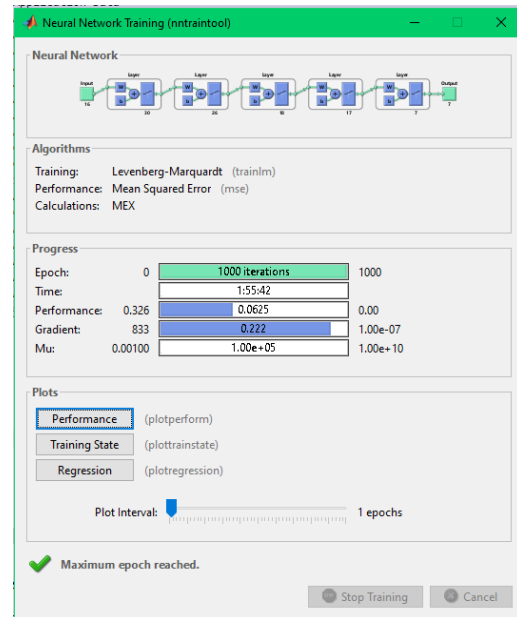
function beansPruebaB
load('Frijoles\Frijoles8.txt')
[m,n] = size(frijoles8);
X = frijoles8(:,1:16)';
t = [];
for i=1:m
    if(frijoles8(i,17)==0) t=[t [1;0;0;0;0;0;0;0]];
    else if(frijoles8(i,17)==1) t = [t,[0;1;0;0;0;0;0;0]];
        else if(frijoles8(i,17)==2)t = [t,[0;0;1;0;0;0;0;0]];
            else if(frijoles8(i,17)==3)t = [t,[0;0;0;1;0;0;0;0]];
                else if(frijoles8(i,17)==4)t = [t,[0;0;0;0;1;0;0;0]];
                    else if(frijoles8(i,17)==5)t = [t,[0;0;0;0;0;1;0;0]];
                        else if(frijoles8(i,17)==6) t = [t,[0;0;0;0;0;0;1;0]];
                            end
                        end
                    end
                end
            end
        end
    end
end
load beansTrain.mat
Y = sim(RNEntrenada, X);
tt = [];
for j=1:m
    [vm,pos]=max(Y(:,j));
    if(pos==1) tt=[tt [1;0;0;0;0;0;0;0]];
    else if(pos==2) tt = [tt [0;1;0;0;0;0;0;0]];
        else if(pos==3) tt = [tt [0;0;1;0;0;0;0;0]];
            else if(pos==4) tt = [tt [0;0;0;1;0;0;0;0]];
                else if(pos==5) tt = [tt [0;0;0;0;1;0;0;0]];
                    else if(pos==6) tt = [tt [0;0;0;0;0;1;0;0]];
                        else if(pos==7) tt = [tt [0;0;0;0;0;0;1;0;0]];
                            else if(pos==8) tt = [tt [0;0;0;0;0;0;0;1]];
                                end
                            end
                        end
                    end
                end
            end
        end
    end
end
%comparar Aciertos
aciertos = 0;
for i=1:m
    [a,pos1] = max(tt(:,i));[aa,pos2] = max(tt(:,i))
    if (pos1 == pos2)
        aciertos=aciertos+1;
    end
end
porcentaje = (aciertos/m)*100
save beansPrueba porcentaje
end

```



porcentaje =

33.1371



porcentaje =

30.1354

Porcentaje Prueba

porcentaje =

28.3639

Lm, 20,20,7 - 32.9017%

Oss las capas aumentaba, pero no pasaba del 19% en entrenamiento, lo mismo pasaba con cgf y cgp

Bfg tiene muy poco porcentaje 3%

Los métodos bfg, cgf,cgp, oss que se probaron con distintas capas y números de neuronas para encontrar el mejor resultado, el porcentaje era muy bajo del 20% y en ciertos casos más rápido que lm, Con lm obteníamos con 2 capas de 10,10 obteníamos resultados rápidos, pero de muy bajo porcentaje, es por eso que decidí incrementar el número de neuronas y capas, el primer resultado vi que aumento el porcentaje muy poco, por lo que aumente el numero de neuronas a 30 30 30 30 7, sin embargo, después de un par de horas no se había avanzado en el proceso como se pensaba.

Se realizó los cálculos de cuánto podría tardar en resolver entre el número de iteraciones y el tiempo que llevaba, dándonos un resultado de 54 horas aproximadamente para terminarse, como el tiempo era demasiado no lo consideramos, cambie las capas y neuronas a uno que lográbamos calcular un tiempo estimado de 16 horas, por lo que el método lm aumentaba muy considerablemente el tiempo, cuando cambie las capas y neuronas un poco más bajas para empezar a tener resultados, las pruebas que realizamos varias veces costaban alrededor de 2 horas aproximadamente. Pero teníamos resultados que no pasaban del 36%.

La última prueba que se realizó. [30 26 18 17 7] Eran casi 2 horas de entrenamiento, con un 30% en entrenamiento y un 28% en las pruebas.

Lo que pudimos observar es que tal vez aumentando el número de capas y neuronas podría aumentar el porcentaje, pero por resultados anteriores no tenemos la seguridad que un entrenamiento muy costoso en tiempo nos de los resultados más óptimos. Por lo que decidí dejarlo en la última prueba.

Debo admitir que para este punto fue un poco agobiante esperar mucho tiempo para tener resultados que no era lo que se esperaban.

Red Neuronal Base Radial

Con un numero de neuronas pequeño daba porcentajes cerca del 14% con 10, 15.65% con 100, Después de muchos minutos con 500 pasaba a 19%

```
porcentaje =  
  
19.3496  
  
>> anr
```

Red Neuronal Probabilística

Código de Entrenamiento

```
function beansProba  
XA = load('Frijoles\frijolesA.txt')  
[m,n] = size(XA);  
X = XA(:,1:16)'; t=XA(:,17)';  
t = t + 1;  
tt = ind2vec(t);  
RN = newpnn(X,tt,0.001);  
Ya = sim(RN,X);  
Y = vec2ind(Ya);  
ac = 0;  
for i=1:m  
    if round(Y(i)) == t(i)  
        ac = ac+1;  
    end  
end  
porcentaje = (ac/m)*100  
  
save beansProbaTrain RN  
save beansProbaPorci porcentaje
```

```
porcentaje =  
  
100
```

Código de Pruebas

```
function beansProbaPrueba  
XA = load('Frijoles\frijolesB.txt')  
[m,n] = size(XA);  
X = XA(:,1:16)'; t=XA(:,17)';  
t = t + 1;  
tt = ind2vec(t);  
load beansProbaTrain.mat  
Ya = sim(RN,X);  
Y = vec2ind(Ya);  
ac = 0;  
for i=1:m  
    if round(Y(i)) == t(i)  
        ac = ac+1;  
    end  
end  
porcentajePrueba = (ac/m)*100  
  
save beansProbaPorciPrueba porcentajePrueba
```

```
porcentajePrueba =  
  
15.0110
```


[Skin]

Para esta ocasión bajamos los datos a 12k aproximadamente ya que Matlab no aceptaba los 122k datos

Red Neuronal Multicapa

Cantidad de capas: 5

Neuronas por capa: 10,10,10,10,2

Funciones de activación: logsig

Métodos de entrenamiento: lm

Código de Entrenamiento

```
function skinsMN
load('Skins\A.txt')

[m,n] = size(A);
X = A(:,1:3)';

t = [];
for i=1:m
    if(A(i,4)==1) t=[t [1;0]];
    else if(A(i,4)==2) t = [t,[0;1]];
    end
end

RNNuNet = newff(minmax(X),[10,10,10,10,2],['logsig','logsig','logsig','logsig','logsig'],'trainlm');
RNEntrenada.trainParam.epochs = 100000;
RNEntrenada.trainParam.min_grad = 1e-27;
RNEntrenada.trainParam.show = 1;
RNEntrenada.trainParam.time = 600;
RNEntrenada = train(RNNuNet,X,t);
Y = sim(RNEntrenada, X);

tt = [];
for j=1:m
    [vm,pos]=max(Y(:,j));
    if(pos==1) tt=[tt [1;0]];
    else if(pos==2) tt = [tt [0;1]];
    end
end

%comparar Aciertos
aciertos = 0;
for i=1:m
    [a,pos1] = max(tt(:,i));[aa,pos2] = max(Y(:,i))
    if (pos1 == pos2)
        aciertos = aciertos+1;
    end
end

porcentaje = (aciertos/m)*100

save skinTrain RNEntrenada
save skinTrainPorciento porcentaje
```

Código de Prueba

```
function skinsPruebaMN
load('Skins\skinB.txt')

[m,n] = size(skinB);
X = skinB(:,1:3)';

t = [];
for i=1:m
    if(skinB(i,4)==1) t=[t [1;0]];
    else if(skinB(i,4)==2) t = [t,[0;1]];
    end
end

load skinTrain.mat
Y = sim(RNEntrenada, X);

tt = [];
for j=1:m
    [vm,pos]=max(Y(:,j));
    if(pos==1) tt=[tt [1;0]];
    else if(pos==2) tt = [tt [0;1]];
    end
end

%comparar Aciertos
aciertos = 0;
for i=1:m
    [a,pos1] = max(tt(:,i));[aa,pos2] = max(Y(:,i))
    if (pos1 == pos2)
        aciertos = aciertos+1;
    end
end

porcentajePrueba = (aciertos/m)*100
porcentajePrueba

save skinPruebaPorciento porcentajePrueba
```

porcentajePrueba =

94.0417

Lm nos dio el mejor resultado por encima de los demás métodos de activación.

Red Neuronal Base Radial

Código de entrenamiento

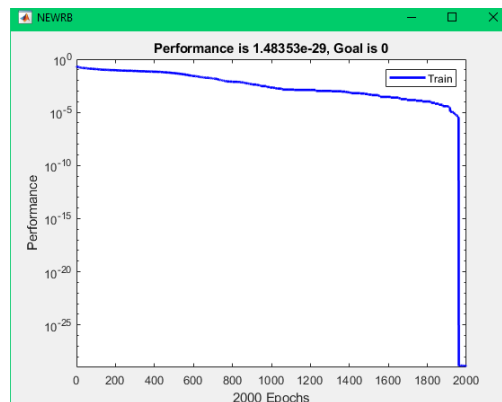
```
function fertilidadRadial
XA = load('Skins\A - copia.txt')
[m,n] = size(XA);
X = XA(:,1:3)'; t=XA(:,4)';
RN = newrb(X,t,0.0,1.0,2100,50);
Y = sim(RN,X);
ac = 0;
for i=1:m
    if round(Y(i)) == t(i)
        ac = ac+1;
    end
end
porcentaje = (ac/m)*100

save skinRadialTrain RN
save skinRadialPorcentaje porcentaje
```

Código de Ejecución

```
function fertilidadRadial
XA = load('Skins\skinB.txt')
[m,n] = size(XA);
X = XA(:,1:3)'; t=XA(:,4)';
load skinRadialTrain.mat
Y = sim(RN,X);
ac = 0;
for i=1:m
    if round(Y(i)) == t(i)
        ac = ac+1;
    end
end
porcentajePrueba = (ac/m)*100

save skinRadialPorcentajePrueba porcentajePrueba
```



Red Neuronal Probabilística

Código de Entrenamiento

```
function skinProba
XA = load('Skins\A.txt');
[m,n] = size(XA);
X = XA(:,1:3)'; t=XA(:,4)';
t = t + 1;
tt = ind2vec(t);
RNp = newpnn(X,tt,0.1);
Ya = sim(RNp,X);
Y = vec2ind(Ya);
ac = 0;
for i=1:m
    if round(Y(i)) == t(i)
        ac = ac+1;
    end
end
porcentaje = (ac/m)*100

save skinProTrain RNp
save skinProTrainPorcentaje porcentaje
```

```
>> skinProbab
```

```
porcentaje =
```

```
100
```

Código de Prueba

```
function skinProbaPrueba
XA = load('Skins\skinB.txt');
[m,n] = size(XA);
X = XA(:,1:3)'; t=XA(:,4)';
t = t + 1;
tt = ind2vec(t);
load skinProTrain.mat
RNp = load('skinProTrain.mat');
Ya = sim(RNp,X);
Y = vec2ind(Ya);
ac = 0;
for i=1:m
    if round(Y(i)) == t(i)
        ac = ac+1;
    end
end
porcentajePrueba = (ac/m)*100

save skinProTrainPorcentajePrueba porcentajePrueba
```

No dejaba usar la red entrenada, me daba error “Sin memoria”

```
>> skinProbaPrueba
Out of memory.

Error in nndata (line 77)
    x(i,1) = value + zeros(elements(i),samples,'like',value);

Error in network/sim>simDataCellofMatrix (line 545)
    Ai = nndata(nn.layer_sizes(net),Q,net.numLayerDelays,zeros('like',arrayType));

Error in network/sim>simData (line 369)
    [err,net,X,Xi,Ai,T,EW,Q,TS] = simDataCellofMatrix(net,X,Xi,Ai,T,EW);

Error in network/sim (line 268)
    [data,err] = simData(net,X,Xi,Ai,T,EW);

Error in skinProbaPrueba (line 8)
    Ya = sim(RNp,X);

Related documentation
```