**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**"Уфимский государственный авиационный технический университет"**

**Кафедра** Высокопроизводительных вычислительных технологий и систем

**Дисциплина:** Интеллектуальные системы

**Отчет по практической работе № 3**

**Тема:** «Нейронная сеть для распознавания трех нелинейно разделимых образов»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Группа ПМ-453 | Фамилия И.О. | Подпись | Дата | Оценка |
| Студент | Шамаев И.Р. |  |  |  |
| Принял | Казакова Т.Г. |  |  |  |

**Уфа 2022**

**Программа**

m=1000; % Число обучающих примеров

h=2; % Параметр обучения в методе градиента

a=1; % Параметр функции активации

w1=0.2\*rand(2,3)-0.1; % Начальные случайные весовые коэф. 1-го слоя

w2=0.2\*rand(3,3)-0.1; % Начальные случайные весовые коэф. 2-го слоя

for k=1:m % Процедура обучения нейронной сети

p=rand;q=rand;r=rand; % Случайный выбор обучающего примера

if p<0.333 x=[1;2\*q+1;2\*r+7];d=[1;0;0];

else if p<0.667 x=[1;2\*q+4;2\*r+4];d=[0;1;0];

else x=[1;2\*q+7;2\*r+1];d=[0;0;1];

end;

end;

u1=w1\*x;

y1=[1;1/(1+exp(-a\*u1(1)));1/(1+exp(-a\*u1(2)))];

v1=w2\*y1;

z=[1/(1+exp(-a\*v1(1)));1/(1+exp(-a\*v1(2)));1/(1+exp(-a\*v1(3)))];

for i=1:3

e(i)=d(i)-z(i);

df2(i)=a\*z(i)\*(1-z(i));

for j=1:3

dw2(i,j)=e(i)\*df2(i)\*y1(j);

end

end

for i=1:2

df1(i)=a\*y1(i+1)\*(1-y1(i+1));

sum=0;

for j=1:3

sum=sum+e(j)\*df2(j)\*w2(j,i+1);

dw1(i,j)=df1(i)\*x(j)\*sum;

end

end

w1=w1+h\*dw1;

w2=w2+h\*dw2;

end

m1=300; % Число итераций для тестирования

for k=1:m1 % Процедура тестирования нейронной сети

p=rand;q=rand;r=rand;

if p<0.333

x=[1;2\*q+1;2\*r+7];

else

if p<0.667

x=[1;2\*q+4;2\*r+4];

else x =[1;2\*q+7;2\*r+1];

end;

end

u1=w1\*x;

y1=[1;1/(1+exp(-a\*u1(1)));1/(1+exp(-a\*u1(2)))];

v1=w2\*y1

z=[1/(1+exp(-a\*v1(1)));1/(1+exp(-a\*v1(2)));1/(1+exp(-a\*v1(3)))];

rx(k)=y1(2); ry(k)=y1(3);

end

g1=[1 3 3 1]; g2=[4 6 6 4]; g3=[7 9 9 7];

s1=[9 9 7 7]; s2=[6 6 4 4]; s3=[3 3 1 1];

subplot(1,2,1);

hold on;

fill(g1,s1,'y',g2,s2,'y',g3,s3,'y');

line([0 8], [-w1(1,1)/w1(1,3); (-w1(1,1)-8\*w1(1,2))/w1(1,3);],'color','b');

line([0 8], [-w1(2,1)/w1(2,3); (-w1(2,1)-8\*w1(2,2))/w1(2,3);],'color','g');

grid on;

subplot(1,2,2);

hold on;

line([0 1], [-w2(1,1)/w2(1,3); (-w2(1,1)-w2(1,2))/w2(1,3);],'color','k');

line([0 1], [-w2(2,1)/w2(2,3); (-w2(2,1)-w2(2,2))/w2(2,3);],'color','b');

line([0 1], [-w2(3,1)/w2(3,3); (-w2(3,1)-w2(3,2))/w2(3,3);],'color','g');

plot(rx,ry,'.');

grid on;

**Результат**

Программа состоит из двух частей, Первая часть моделирует процедуру обучения нейронной сети. Вторая — проводит тестирование обученной сети.

В данной задаче число обучающих примеров (точек из множеств L1, L2 и L3) предполагалось неограниченным. В результате для обучения сети потребовалось 1000 примеров. Процедура обучения признавалась успешной и заканчивалась, когда сеть правильно устанавливала границы раздела.

Обученная нейронная сеть подвергалась затем процедуре тестирования на примере 300 объектов, выбранных случайным образом из классов L1, L2 и L3. Свойства нейронов первого слоя отражены на рис. 1, на левом графике. Здесь показаны классы L1, L2, L3 и границы раздела первого и второго нейронов. Правый график показывает, как первый слой нейронов преобразует исходные линейно неразделимые классы объектов в линейно разделимые. Правый график на рисунке 1 отражает свойства нейронной сети с гладкими функциями активации, в которой преобразование приводит к «размытым» классам. Окончательное разделение обеспечивает второй слой нейронов. Распознавание объекта происходит по превышению одним из сигналов v1, v2 или v3 уровня 0,5.

На практике число обучающих примеров является ограниченным. Все примеры в виде векторов Х и D оформляются в пакет, который подается на нейронную сеть. Весовые коэффициенты корректируются после каждого примера. Прохождение пакета примеров и коррекция весовых коэффициентов составляют эпоху в процедуре обучения. Для успешного обучения требуется, как правило, много эпох. Внутри эпохи обучающие примеры могут занимать привычные места, а могут перемешиваться случайным образом. Обучение сети распознаванию образов заканчивается на той эпохе, в которой все примеры правильно классифицированы.



Рисунок 1. Границы раздела обученной сети программы

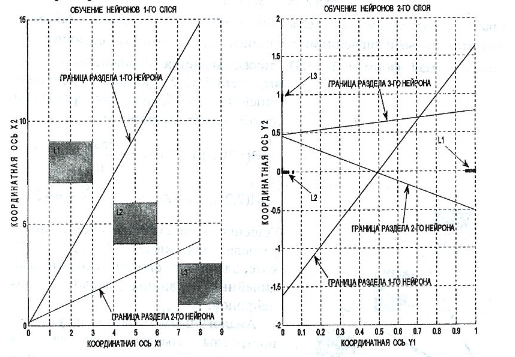


Рисунок 2. Пример границы раздела обученной сети