

Лабораторная работа

Радиальные базисные сети

Цель работы – изучить архитектуру и процесс обучения радиальных базисных сетей.

Содержание:

1. Изучение архитектуры и обучение радиальных базисных сетей в MATLAB
2. Создание базисной сети с нулевой ошибкой.
3. Создание нейронной сети радиального типа для аппроксимации поверхности.
4. Создание графиков радиальных базисных функций, определение взвешенной суммы данных функций.

Порядок выполнения заданий

Задание 1. Изучение архитектуры и обучение радиальных базисных сетей в MATLAB.

Радиальная базисная сеть (Radial Basis Function, RBF-сеть) состоит из двух слоев. Первый слой включает нейроны, использующие радиальные функции *radbas*. Функции вычисляют расстояние с помощью функции *dist* между нейронами сети в ходе обучения и входными векторами на основе расстояния (рис. 1).

При этом сведения о расположении нейронов на каждой итерации хранятся в строках матрицы весов $net.IW\{1,1\}$. Второй слой радиальной сети включает нейроны с линейной функцией активации *purelin*.

Для создания RBF-сети применяется функция *newrb*, синтаксис которой приведен ниже:

$$net = newrb(P,T,GOAL,SPREAD,MN,DF),$$

где P – матрица $R \times Q$ из Q входных векторов размерностью R ;

T – матрица $S \times Q$ из Q целевых векторов размерностью S ;

GOAL – максимально допустимая ошибка;

SPREAD – параметр сглаженности радиальных функций;

MN – максимально допустимое количество нейронов в первом слое;

DF – количество нейронов, добавляемых на каждом шаге.

При создании нейронной сети необходимо обязательно указывать первые два параметра, остальные параметры определяются по умолчанию.

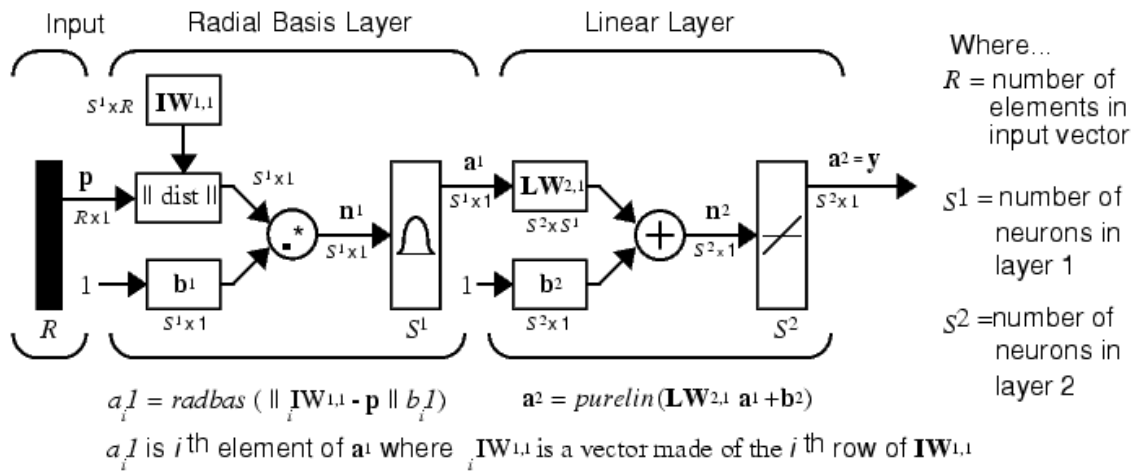


Рис.1. Функции активации при обучении радиальных базисных сетей

Радиальная базисная сеть состоит из двух слоев:

- одного скрытого слоя, который называют слоем радиальных нейронов, или радиальным базисным слоем, включающим S^1 нейронов;
- выходного слоя нейронов, включающего S^2 нейроны с линейной функцией активации, данный слой называют линейным слоем (рис. 1).

Вектор входных данных, как правило, обозначается через p . При вычислении расстояний в блоке $\|dist\|$ используется вектор p и матрица весовых коэффициентов $IW^{1,1}$.

Результатом вычислений блока $\|dist\|$ является вектор, состоящий из S^1 элементов, для которых определяются расстояния между i -м вектором входа p и i -й вектор-строкой $IW^{1,1}$ матрицы весов. Данную вектор-строку называют вектором весов i -го нейрона. Выход блока $\|dist\|$ умножается поэлементно на вектор смещения b^1 и формирует вход аргумента функции активации. В результате, выходные значения первого слоя нейронной сети записываются в виде:

$$a = \text{radbas}(\text{net.prod}(\text{dist}(\text{net.IW}, p), \text{net.b}))$$

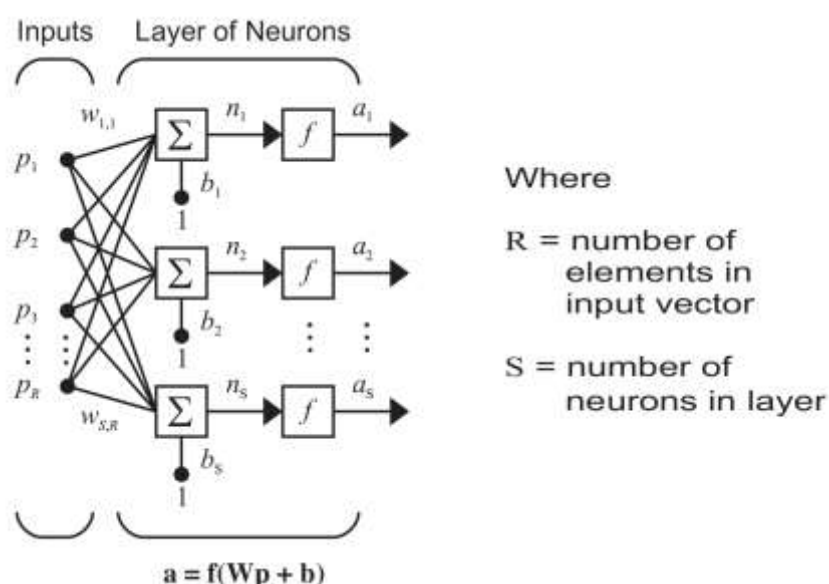
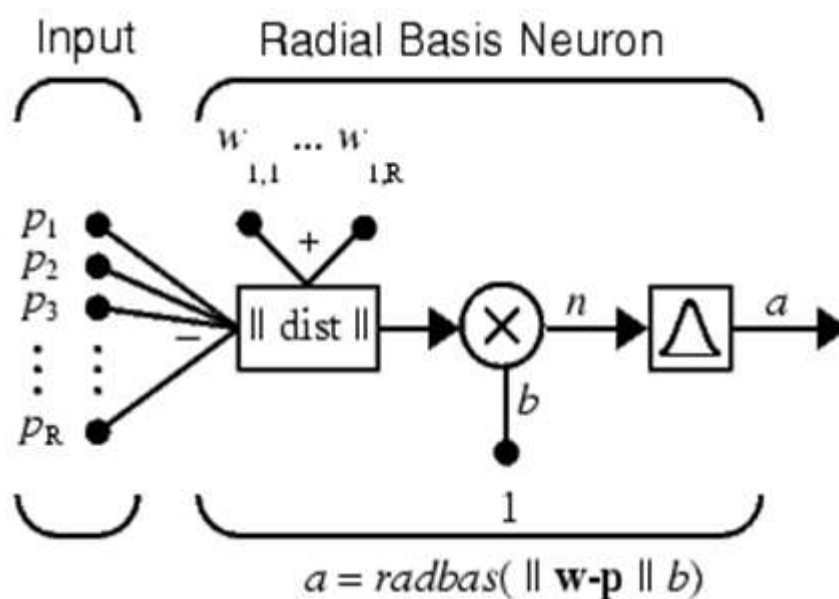


Рис.2. Архитектура радиальной базисной сети

Радиальная базисная функция имеет максимум, равный 1, в том случае, если вход сети равен 0. Радиальный базисный нейрон реагирует подобно индикатору, формирующему на выходе значение 1, если вход сети **буден** идентичен вектору весовых коэффициентов. В том случае если, расстояние между векторами входов и весами уменьшается, то выход радиальной базисной функции увеличивается. В том случае, если только один радиальный базисный нейрон имеет выходное значение, равное 1, а все остальные нейроны имеют выходы,

равные или очень близкие к 0, то выходные значения на втором слое сети будут равны весам активного выходного нейрона. Это редкий случай, обычно на выходное значение сети оказывают влияние несколько нейронов с различными весовыми коэффициентами.

Смещение нейронной сети используется для корректирования чувствительности нейрона с помощью функции *radbas*. Например, если нейрон обладает смещением равным 0,1, то его выход будет равен 0,5 для любого вектора входа p и вес нейрона w при расстоянии между векторами, равном 10.333 составит $10.833/b$.

Задание 2. Создание базисной сети с нулевой ошибкой.

Для построения радиальных базисных сетей предназначены М-функции *newrbe* и *newrb*. Первая функция позволяет построить радиальную базисную сеть с нулевой ошибкой, вторая позволяет управлять количеством нейронов скрытого слоя. Функция *newrbe*, которая вызывается следующим образом:

$$\text{net} = \text{newrbe}(P, T, \text{SPREAD})$$

Входными аргументами функции *newrbe* являются массивы входных векторов P и целей T , а также параметр влияния SPREAD . Она возвращает радиальную базисную сеть с такими весами и смещениями, что ее выходы точно равны целям T . Функция *newrbe* создает столько нейронов радиального базисного слоя, сколько имеется входных векторов в массиве P и устанавливает веса первого слоя равными P' .

При этом смещения устанавливаются равными $0.8326/\text{SPREAD}$. Это означает, что уровень перекрытия радиальных базисных функций равен 0.5 и все входы в диапазоне $\pm \text{SPREAD}$ считаются значимыми. Чем больший диапазон входных значений должен быть принят во внимание, тем большее значение параметра влияния SPREAD должно быть установлено (демонстрационная программа *demorb1*).

Веса второго слоя IW^{21} и смещений b^2 (b) могут быть найдены путем моделирования выходов первого слоя a^1 (A) и последующего решения системы линейных алгебраических уравнений (СЛАУ):

$$[W \ b] * [A; \text{ones}] = T$$

Поскольку известны входы второго слоя A и цели T , а слой линеен, то для вычисления весов и смещений второго слоя достаточно воспользоваться решателем СЛАУ:

$$Wb = T / [P; \text{ones}(1, \text{size}(P,2))]$$

Матрица Wb содержит веса и смещения. Сумма квадратов погрешностей сети всегда равна 0, т.к. имеем задачу с Q уравнениями (пары вход/цель) и каждый нейрон имеет $Q + 1$ переменных (Q весов по числу радиальных базисных нейронов и одно смещение). СЛАУ с Q уравнениями и более чем Q переменными имеет свободные переменные и характеризуется бесконечным числом решений с нулевой погрешностью.

Пример 1. Использование радиальной функции активации в MATLAB при изменении переменной x от -5 до 5 с шагом 0,1.

```
>> x = -5:.1:5;
>> a = radbas(x);
>> plot(x,a)
>> title('Радиальная базисная функция', 'FontSize', 16);
>> xlabel('Входные данные', 'FontSize', 16);
>> ylabel('Выходные данные', 'FontSize', 16);
>> grid on
```

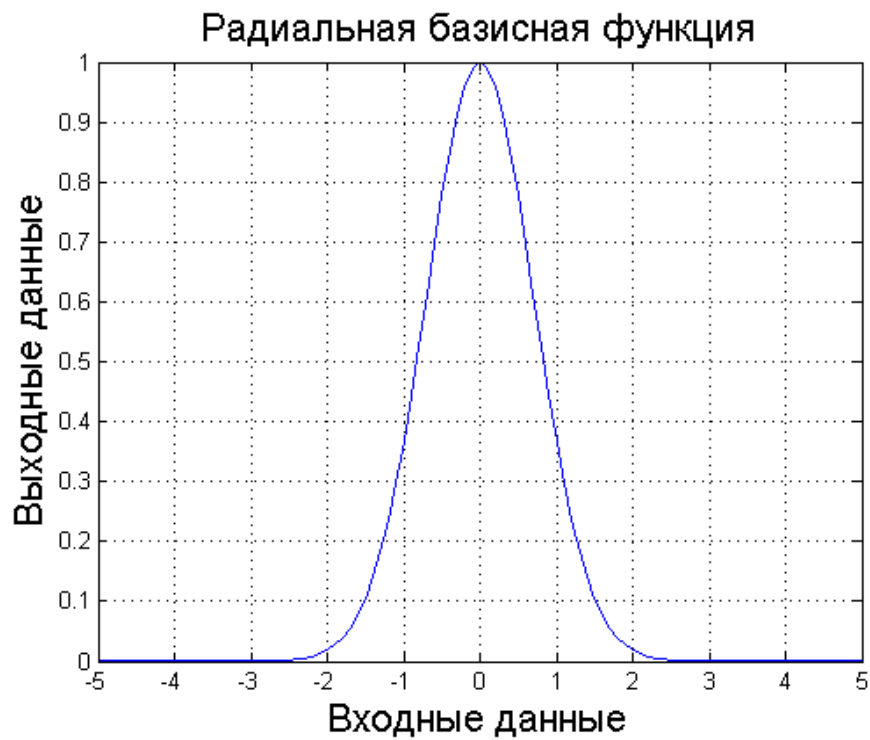


Рис.3. Радиальная базисная функция

Исходные данные для выполнения задания необходимо внести в табл. 1:

P – вектор исходных данных изменяется -2 до 1,4 с шагом 0,2

T – вектор целей вычисляется по формуле

$$T = \cos(P + 0.05 * N) + 0.04, \quad (2)$$

Таблица 1 – Исходные данные для выполнения задания

P	-2	-1,8	-1,6	-1,4	-1,2	-1,0	-0,8	-0,6	-0,4
T									

Продолжение таблицы 1

P	-0,2	0	0,2	0,4	0,6	0,8	1,0	1,2	1,4
T									

Пример.2. Создание радиальной базисной сети с нулевой ошибкой

```
>> P=[0.2 1 1.8 2.6 3.4 4.2 5 5.8 6.6 7.4 8.2]
```

```

>> T=[1.18006  0.74030  -0.02720  -0.65688  -0.76679  -0.29026
0.483662185  1.085519  1.15023  0.63854  -0.13915]
>> plot(P,T,'*r','MarkerSize',4,'LineWidth',2)
hold on
grid on
>> % Создание радиальной базисной сети
net = newrbe(P,T);
% Определение числа нейронов в скрытом слое
>> net.layers.size
%ans = 21
% Моделирование сети с использование входных и целевых данных
обучающей выборки
V = sim(net,P)
ans =
    [11]
    [ 1]
%Полученные значения в результате обучения нейронной сети

V = 1.1801  0.7403 -0.0272 -0.6569 -0.7668 -0.2903  0.4837  1.0855
1.1502 0.6385 -0.1392
% создание графика с исходными данными
>> plot(P,V,'ob','MarkerSize',5, 'LineWidth',2)
% задание вектора промежуточных точек в виде вектора входных дан-
ных
p = [0.6 1.6 2.3 4.0 6.5];
% процесс переобучения нейронной сети
v = sim(net,p)
plot(p,v,'+k','MarkerSize',10, 'LineWidth',2)
grid on
% получение результатов обучения
% построение графика и сетки на графике
v =

    1.0155    0.1722   -0.4663   -0.4509    1.1688

```

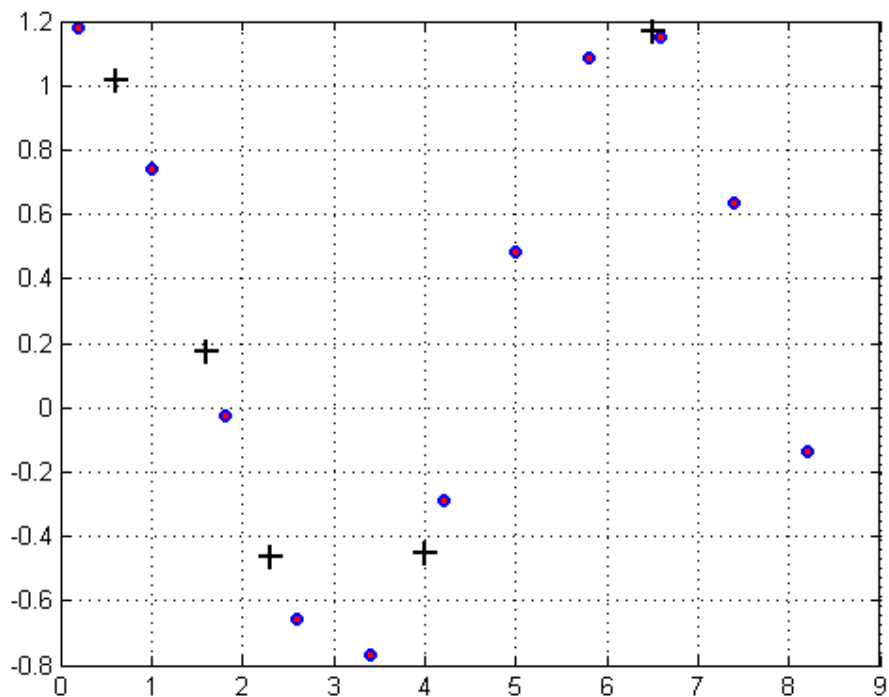


Рис.4. Результаты аппроксимации функции с помощью радиальной базисной сети

Задание 3. Создание нейронной сети радиального типа для аппроксимации поверхности.

Исходные данные для выполнения задания:

$$z = (\cos(x \cdot \sin(y + 0,02 \cdot N) + 0,08 \cdot N)$$

Переменные x, y изменяется на интервале от -3 до 3 с шагом $0,01 \cdot N$ (N – номер варианта задания).

Пример 3. Необходимо создать поверхность, описываемой в виде формулы, для которой переменные x, y изменяется на интервале от -3 до 3 с шагом 0,1:

$$z = (\cos(x \cdot \sin(y + 0,1) + 0,2). \quad (1)$$

Для этого нужно задать исходные данные, создать и обучить сеть, вывести результаты обучения на экран и построить график смоделированной поверхности (рис.2). Программный код MATLAB приведен ниже:

Пример 3. Необходимо создать поверхность, описываемой в виде формулы, для которой переменные x, y изменяются на интервале от -3 до 3 с шагом 0,1:\

\par\singlespacing $z = (\cos(x \cdot \sin(y + 0,1)) + 0.2)$

% определение исходных данных, вектора входных данных для обучения сети

$x = -3:0.1:3;$

% определение вектора выходных данных

% радиальные базисные сети обучаются только с учителем

$y = -3:0.1:3;$

% задание уравнения поверхности от двух переменных

$z = \cos(x \cdot \sin(y + 0.1)) + 0.2;$

% определение матрицы входов и выходов нейронной сети

$P = [x; y]; T = z;$

% создание нейронной сети радиального типа

$net = newrb(P, T);$

% обучение сети в зависимости от выбранной архитектуры

$Y = sim(net, P)$

% построение трехмерной поверхности

$mesh(x, y, z)$

% получение ошибки обучения для каждого нейрона

% и построение поверхности

NEWRB, neurons = 0, MSE = 0.182572

NEWRB, neurons = 2, MSE = 0.101978

NEWRB, neurons = 3, MSE = 0.0165157

NEWRB, neurons = 4, MSE = 0.014556

NEWRB, neurons = 5, MSE = 0.00523354

NEWRB, neurons = 6, MSE = 0.00256723

NEWRB, neurons = 7, MSE = 0.00238156

NEWRB, neurons = 8, MSE = 0.00222853

NEWRB, neurons = 9, MSE = 3.7241e-05

NEWRB, neurons = 10, MSE = 1.64223e-05

NEWRB, neurons = 11, MSE = 8.21751e-06

NEWRB, neurons = 12, MSE = 3.57725e-06

NEWRB, neurons = 13, MSE = 2.89488e-06

NEWRB, neurons = 14, MSE = 2.48158e-07

NEWRB, neurons = 15, MSE = 4.84938e-08

NEWRB, neurons = 16, MSE = 2.15415e-08

NEWRB, neurons = 17, MSE = 1.50764e-08

NEWRB, neurons = 18, MSE = 5.54992e-09

NEWRB, neurons = 19, MSE = 4.55362e-09
NEWRB, neurons = 20, MSE = 4.36135e-11
NEWRB, neurons = 21, MSE = 3.36148e-11

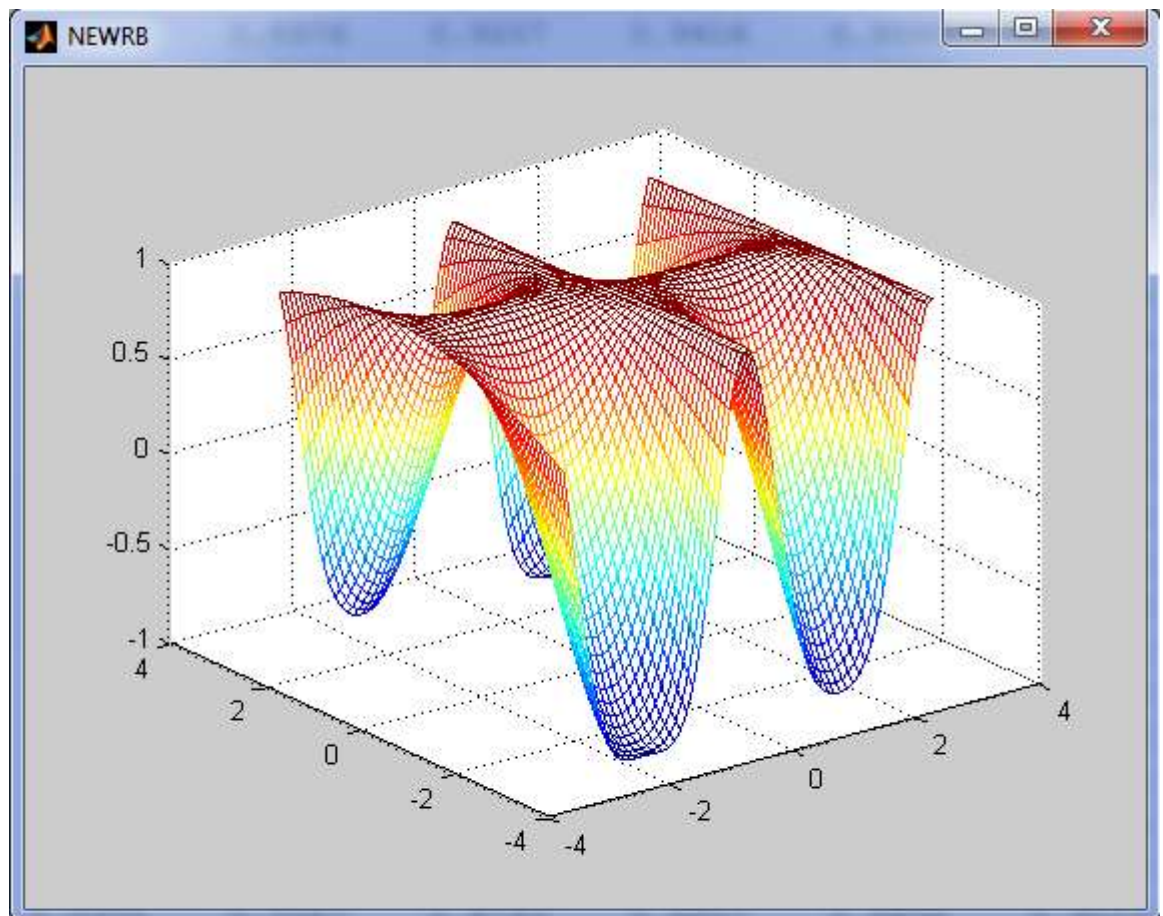


Рис.5. Результат моделирования нейронной сети и построение поверхности функции

Задание 4. Создание графиков радиальных базисных функций, определение взвешенной суммы данных функций.

Исходные данные для выполнения задания (N – номер варианта задания):

```
p = -3*N:0.2:3*N;  
a1 = radbas(p);  
a2 = radbas(p-0.5*N);  
a3 = radbas(p+3*N);  
a = a1 + a2*1 + a3*0.5;
```

Пример 4. Программный код MATLAB для создания графиков радиальных базисных функций.

```

p = -3:0.1:3;
a1 = radbas(p);
a2 = radbas(p-1.5);
a3 = radbas(p+2);
a = a1 + a2*1 + a3*0.5;
plot(p,a1,p,a2,p,a3*0.5,p,a)

```

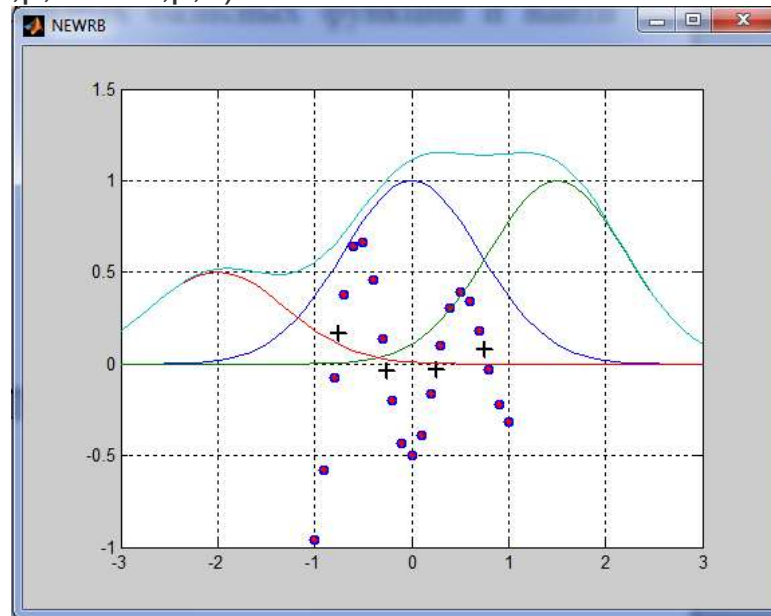


Рис.6. Графики радиальных базисных функций

Для сдачи лабораторной работы:

1. Отчет о выполненной работе должен содержать таблицы исходных данных, программный код, рисунки с результатами обучения.
2. Выводы, список использованной литературы.

Контрольные вопросы:

1. Какую архитектуру имеет нейронная сеть радиального типа?
2. Какие функции используются при обучении радиальной базисной сети?
3. Для решения каких практических задач используется радиальная сеть?
4. Каким образом выполняется обучение радиальной базисной сети с нулевой ошибкой?
5. Какая последовательность обучения радиальной сети?
6. Для каких целей задается смещение сети?
7. Что понимают под радиальными базисными функциями?

8. Какие функции использует радиальный и линейный слои радиальной сети?
9. Каким образом производится обучение радиальных базисных сетей?
10. Какие параметры используются при задании функции newtbe?
11. Обучение радиальных базисных сетей производится с учителем или без учителя?
12. При решении каких задач используются радиальные базисные сети?