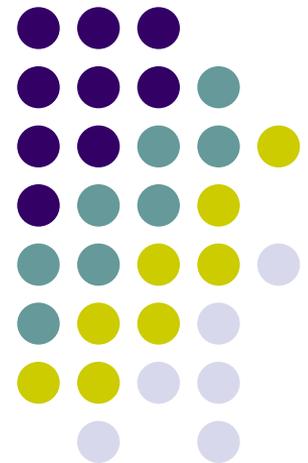
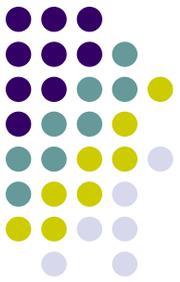


# Сети радиальных функций

Корлякова М.О.  
2016

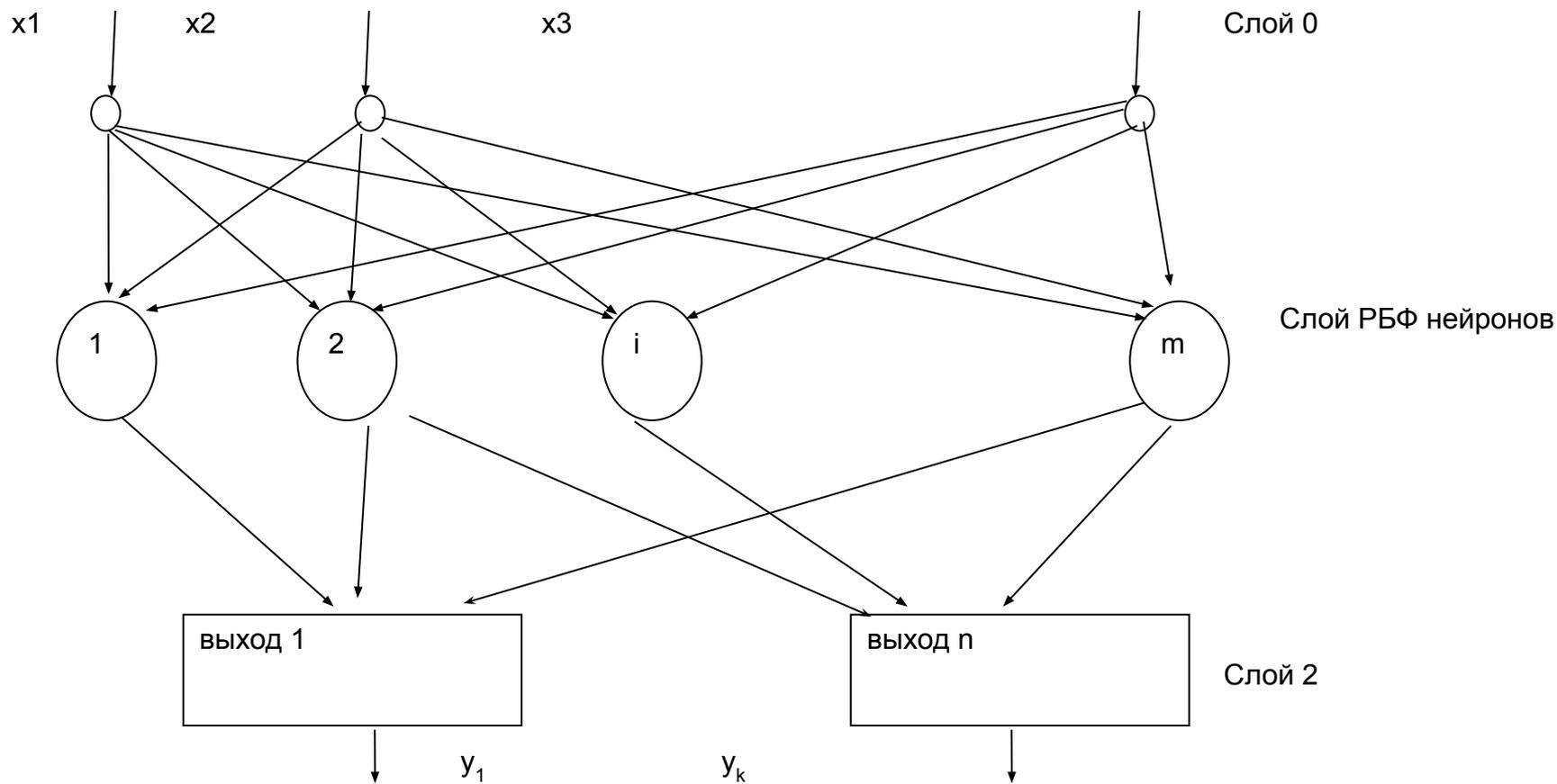


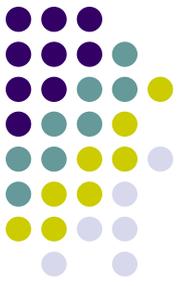


# Принцип

- **Обучение.** Аппроксимация кривой по точкам в пространстве высокой размерности.
- **Обобщение.** Используем многомерную кривую для тестирования (интерполяция).
- Используем специальные функции преобразования для разложения входного образа.

# Архитектура сети РБФ

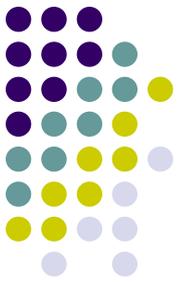




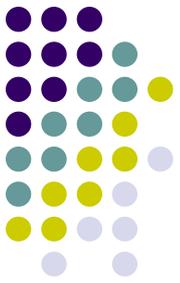
# *Архитектура сети РБФ*

- Обучение с учителем.
- Обучение по соревнованию.
- Слой 0 – рецептивный, слой 2 – линейные нейроны.
- Число входов  $n$ , число выходов совпадает с числом формируемых классов.
- Сеть прямого распространения.

# Аргумент радиальной функции



$$r_{ij} = \sqrt{\sum_{i=0}^{p-1} (x_i - w_{ij})^2}$$



# Подстройка синапсов

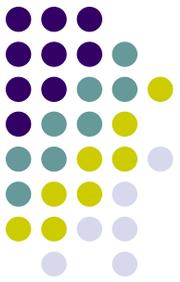
От рецептивного слоя:

- Определить расстояние до образца
- Сдвинуть к образцу

$$C_i(t+1) = C_i(t) + \eta \cdot (\mathbf{X}^k - C_i(t)),$$

От скрытого слоя

$$w_{ij} = w_{ij} + a \cdot y_i \cdot (y_j - d_j)$$



# Создание РБФ-сети

- `newrb(PR, T, goal, spread, MN, DF),`

где `PR` – матрица столбцов входных значений,

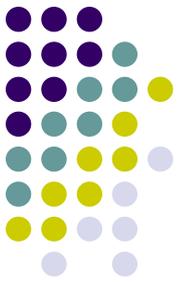
`T` – матрица целевых значений,

`goal` – допустимое значение функционала ошибки,

`spread` – диапазон перекрытия входных значений (размах нейрона или его влияние),

`MN` – максимально-возможное количество нейронов в скрытом слое (по умолчанию равно количеству входов),

`DF` – интервал (количество нейронов), по истечении которого на дисплей выводятся промежуточные результаты обучения.



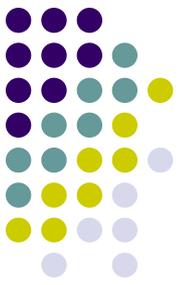
# Пример

**P = 0:3;**

**T = [0.0 2.0 4.1 5.9]; % целевое значение**

Создаем растущую сеть РБФ.

**net = newrb(P,T,0.1);**



# Задача: предсказание

Обучение:

$$x = -10\pi : 0.5 : 10\pi;$$

$$y = 10\sin(x) + 3\cos(2x) + \sin(0.5x) + \cos(10x);$$

$$T = \{(X_i, d_i)\}$$

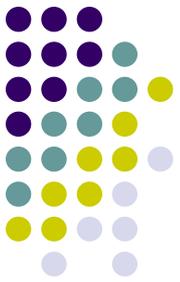
$$X_i = \langle y(x(i)), y(x(i-1)), y(x(i-2)), y(x(i-3)), y(x(i-4)) \rangle$$

$$d_i = y(x(i+1))$$

Тест:

$$x_2 = -10.2\pi : 0.5 : 10.2\pi;$$

$$y_2 = 10\sin(x) + 3\cos(2x) + \sin(0.5x) + \cos(10x);$$



# Задача: предсказание

%заполнение обучающей выборки

```
X=[];d=[];
```

```
for i=1:N
```

```
    p=[y(i:i+4)];
```

```
    X=[X; p];
```

```
    d=[d y(i+5)];
```

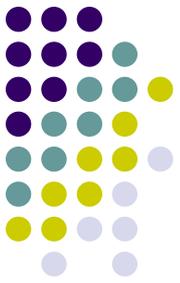
```
end;
```

%обучение

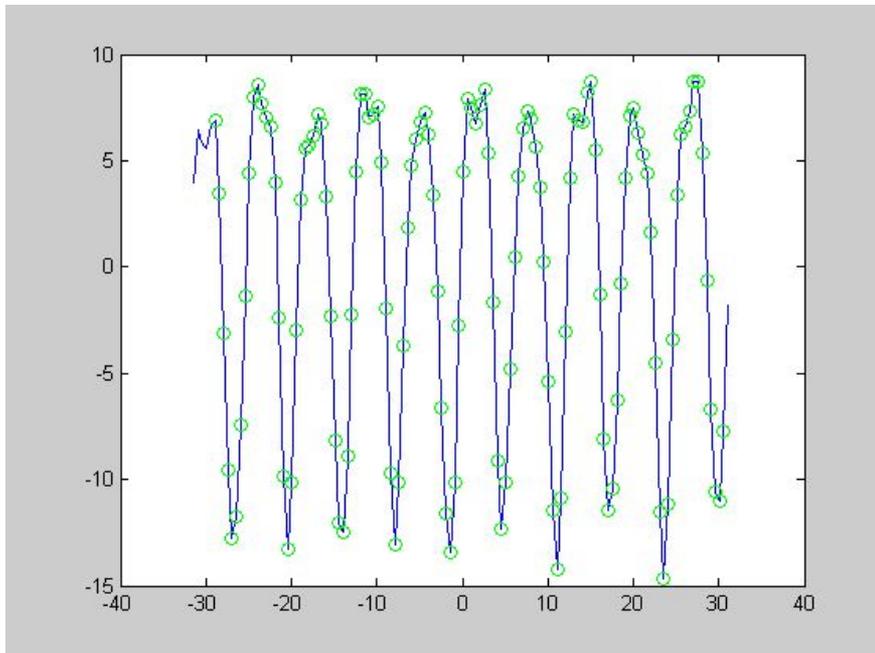
```
net=newrb(X',d);
```

```
y1=sim(net,X');
```

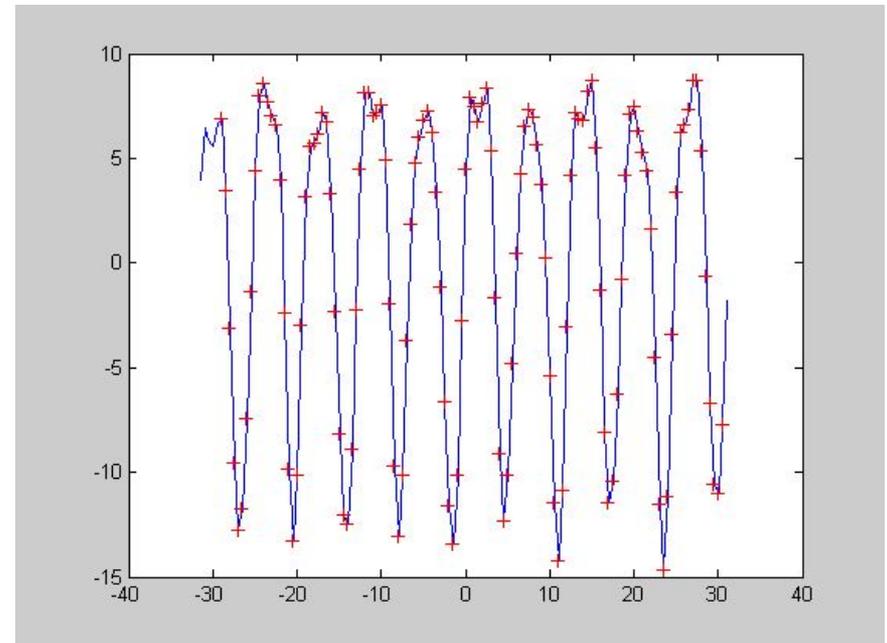
# Задача: предсказание

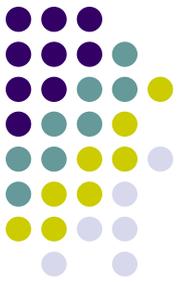


Обучение:



Тест:





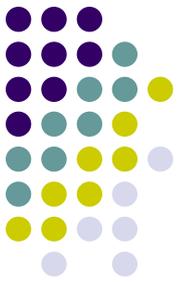
# Классификация

Примеры:

$x = 0$	0	0	0	1	$t = \{1\ 0\}$	$x = 0$	0	0	0	1	$t = \{0\ 1\}$	
	1	0	1	1			1	0	0	0	0	
	0	0	0	1			0	0	0	0	1	
	1	0	0	0			1	0	0	0	0	
	1	1	1	1			1	0	0	0	0	

Обучение:

`net=newrb(P',T');`



# Классификация

$y1(:,1000)$

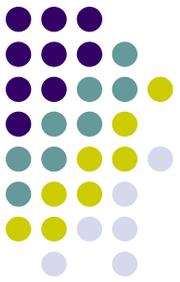
0

1.0000

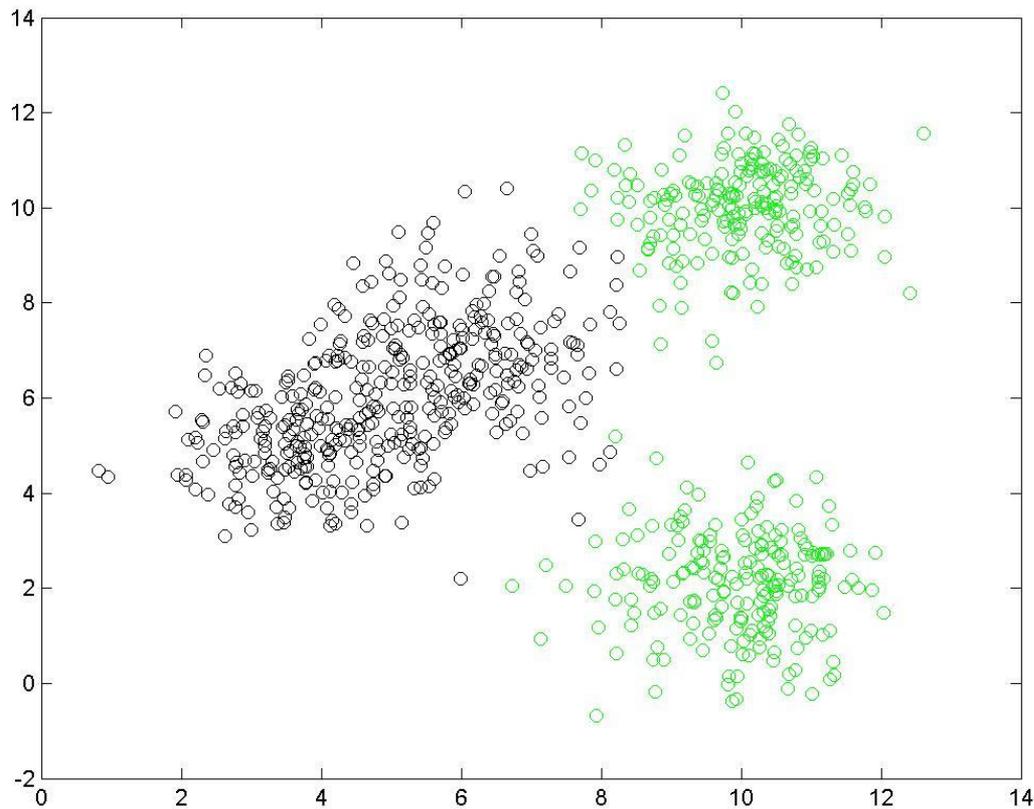
$x1 =$

0	0	0	0	1
1	0	1	0	0
0	0	0	0	1
1	0	0	0	0
1	1	1	1	0

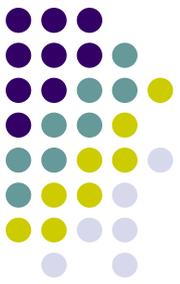
# Результаты обучения RBF



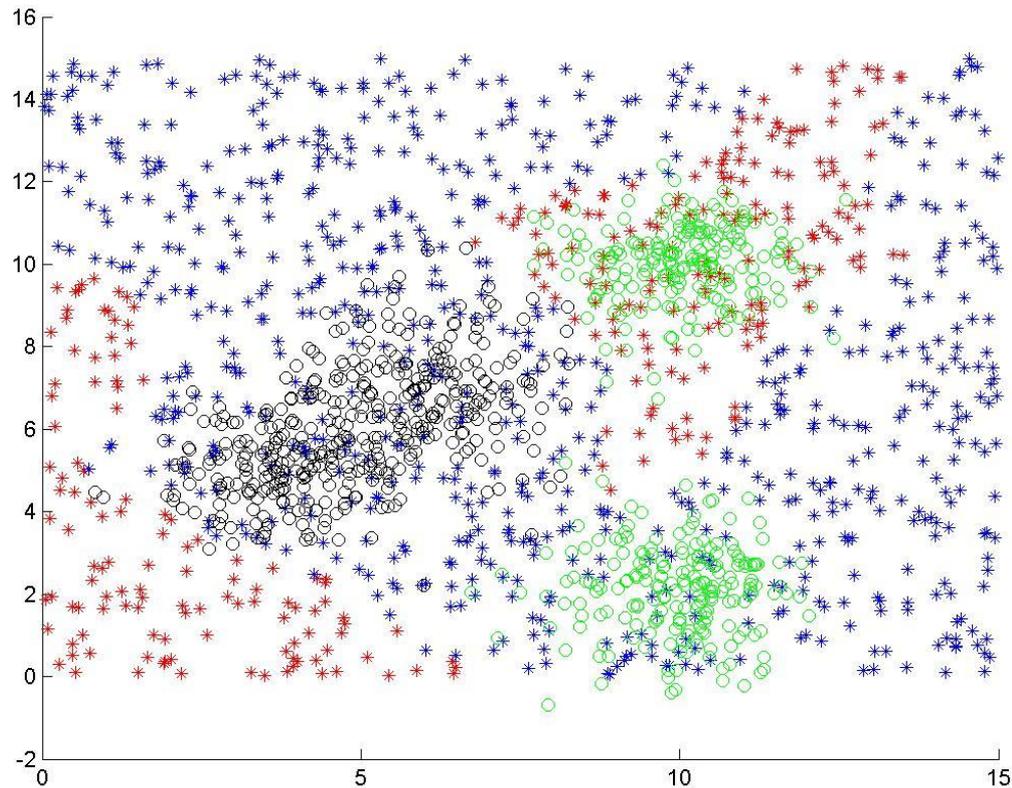
- Исходное множество



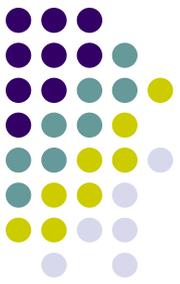
# Результаты обучения RBF



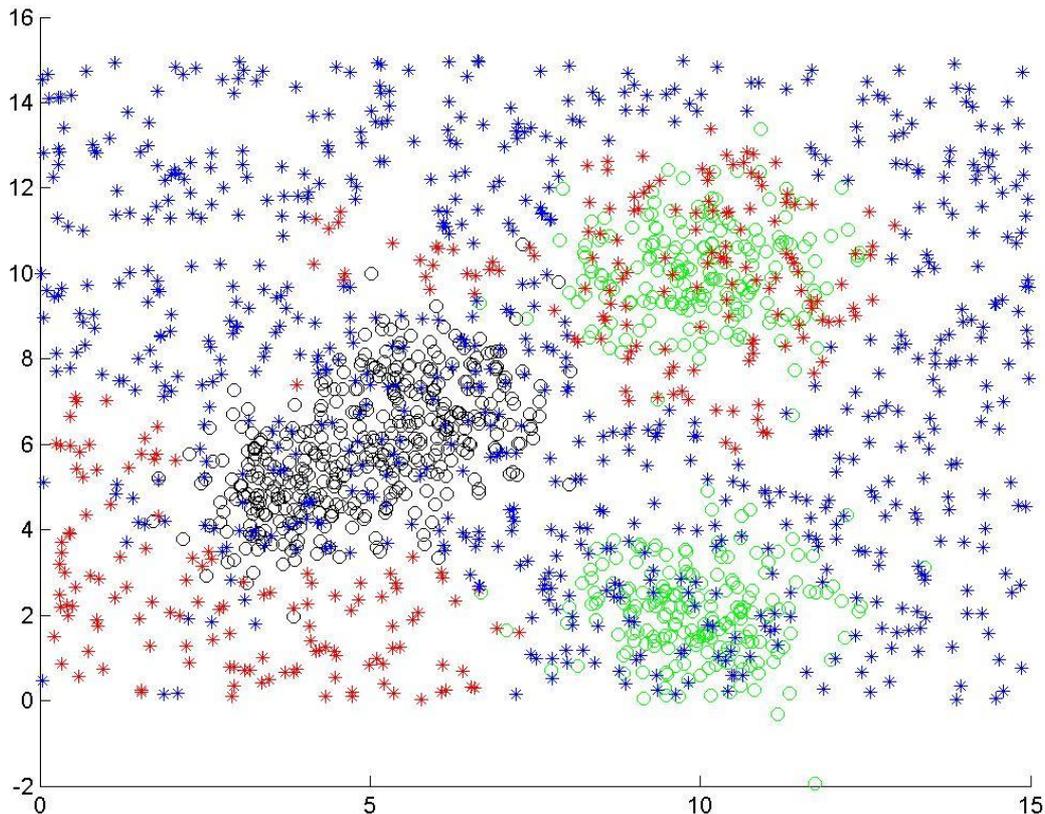
- Результат обучения для 400 нейронов 10



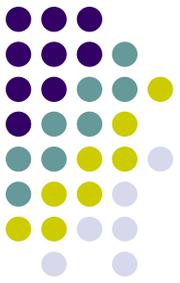
# Результаты обучения RBF



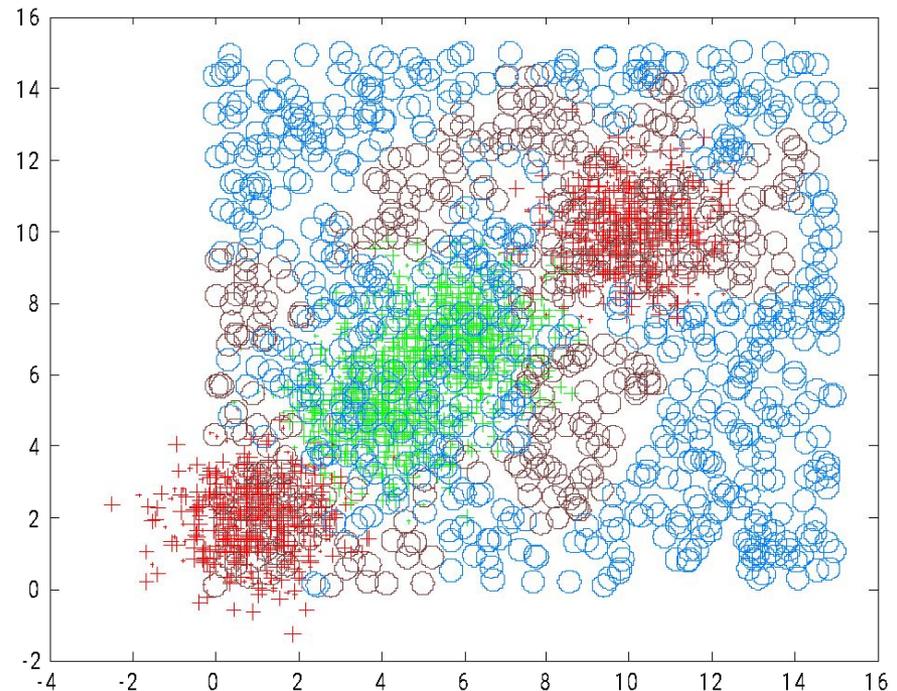
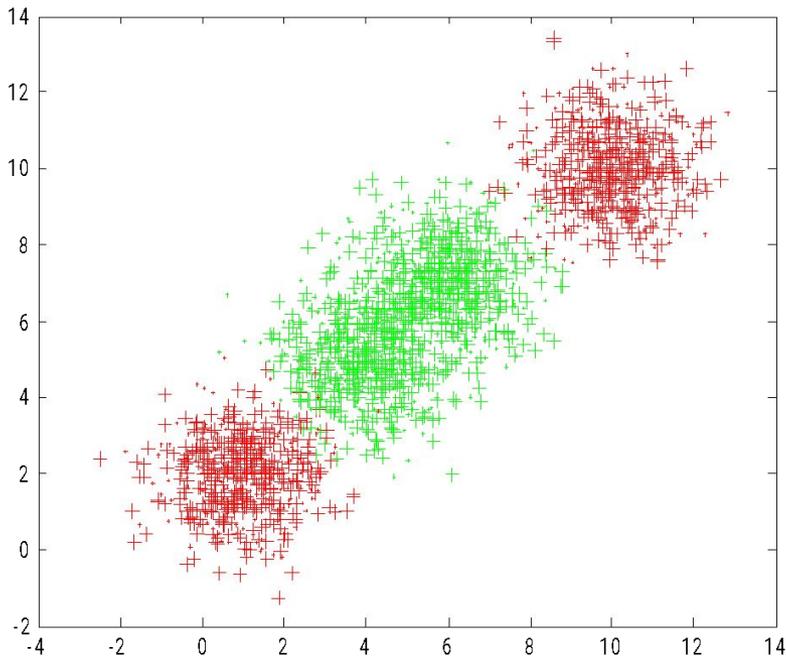
- Результат обучения для 400 нейронов 1



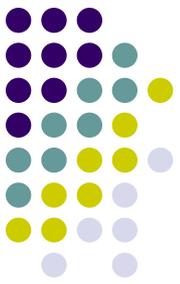
# Результаты обучения RBF



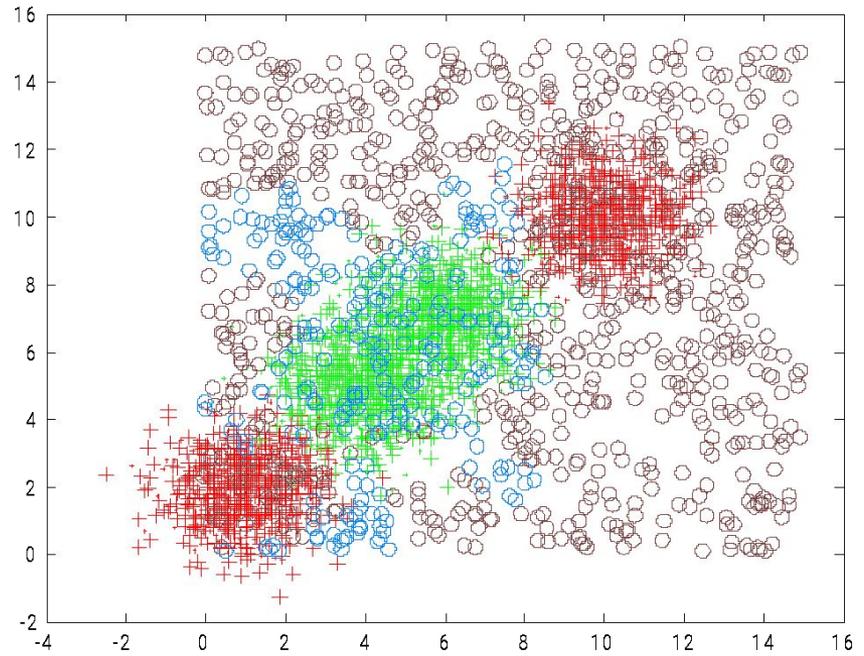
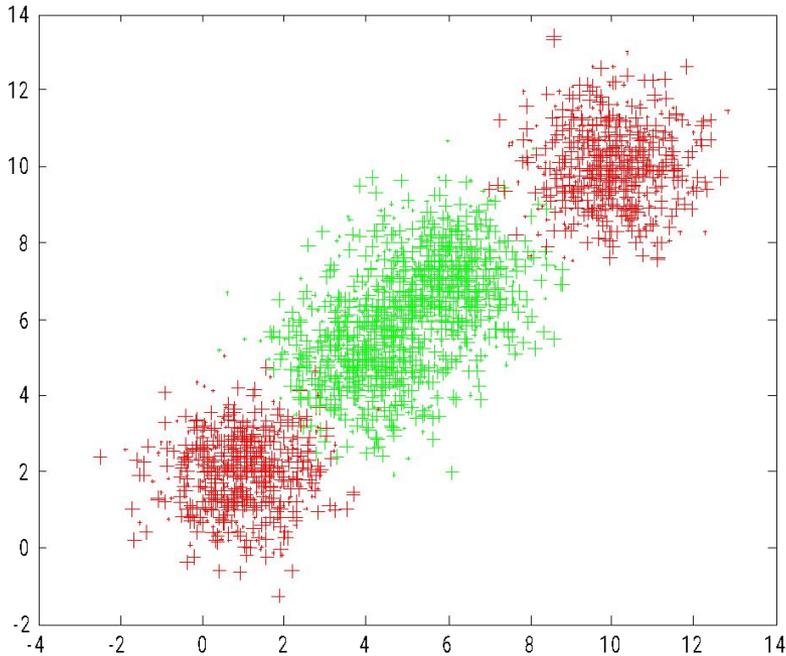
- Результат обучения для 300 нейронов 1



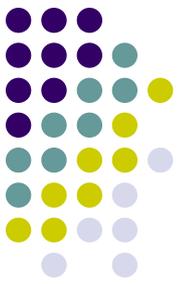
# Результаты обучения RBF



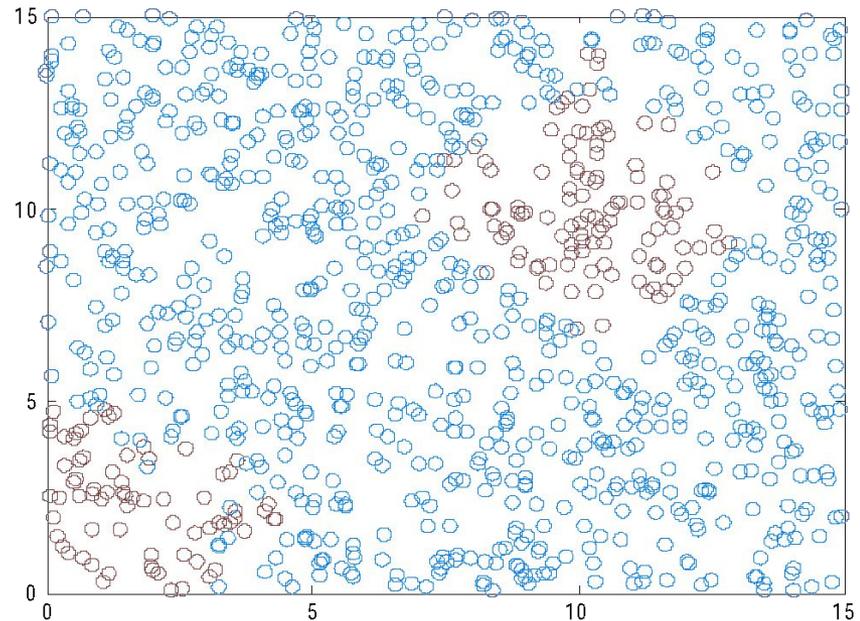
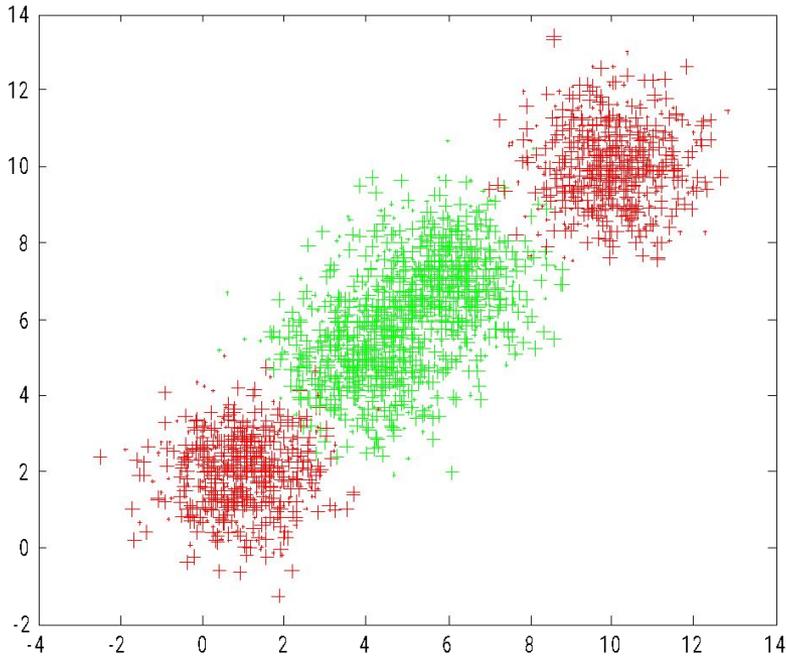
- Результат обучения для 400 нейронов 1



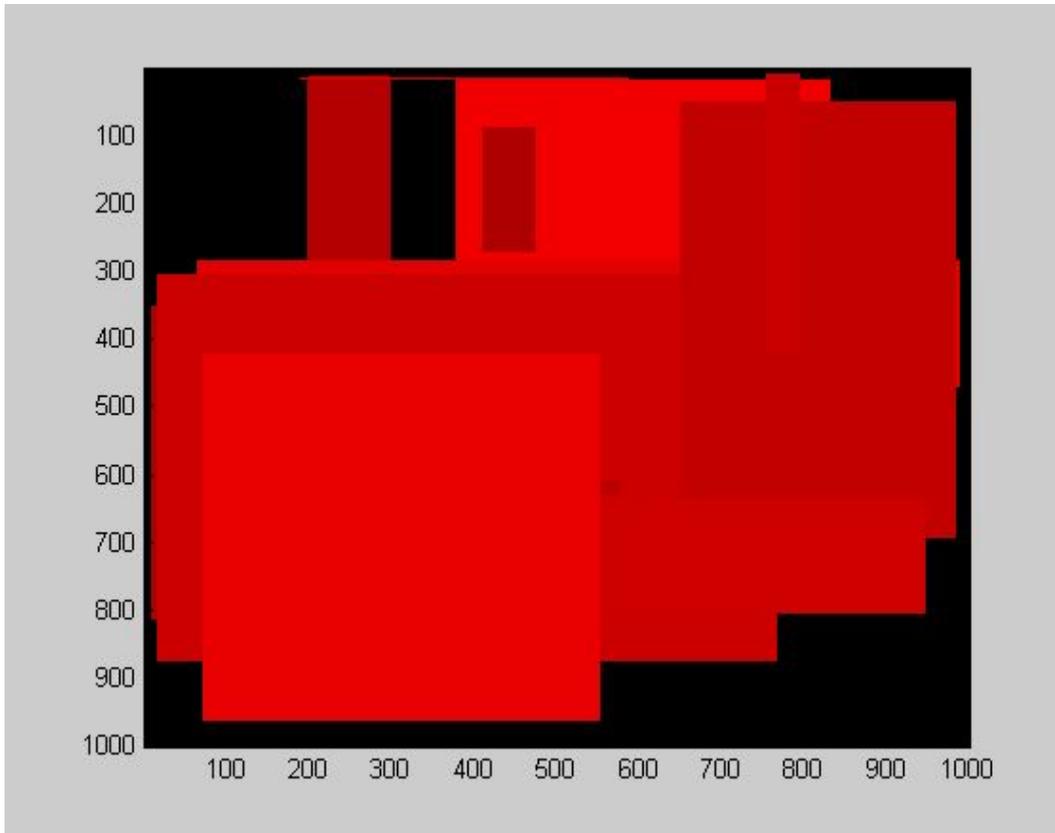
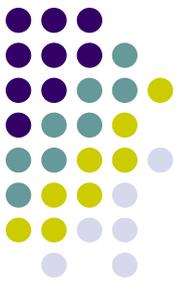
# Результаты обучения RBF



- Результат обучения для 400 нейронов 0.5



# Сегментация изображений

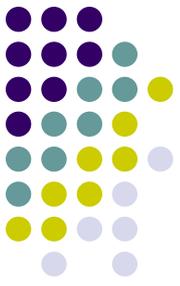


$$T = \{(X_i, d_i)\}$$

$d_n = 5$  – размер  
блока

$$X_i = A(i:i+d_n, j:j+d_n)$$

$d_i = \{1, \text{если}$   
красный,  $-1,$   
если не  
красный}

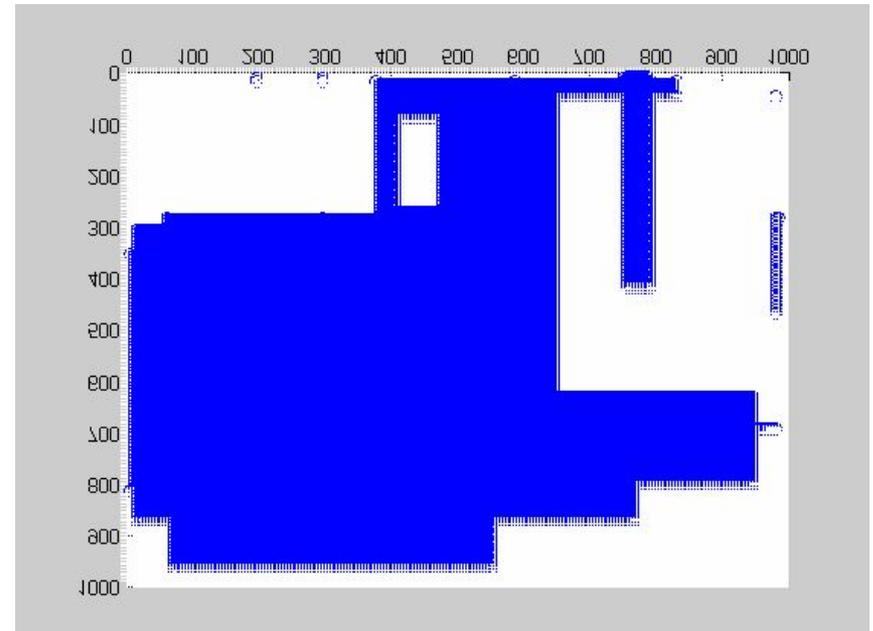
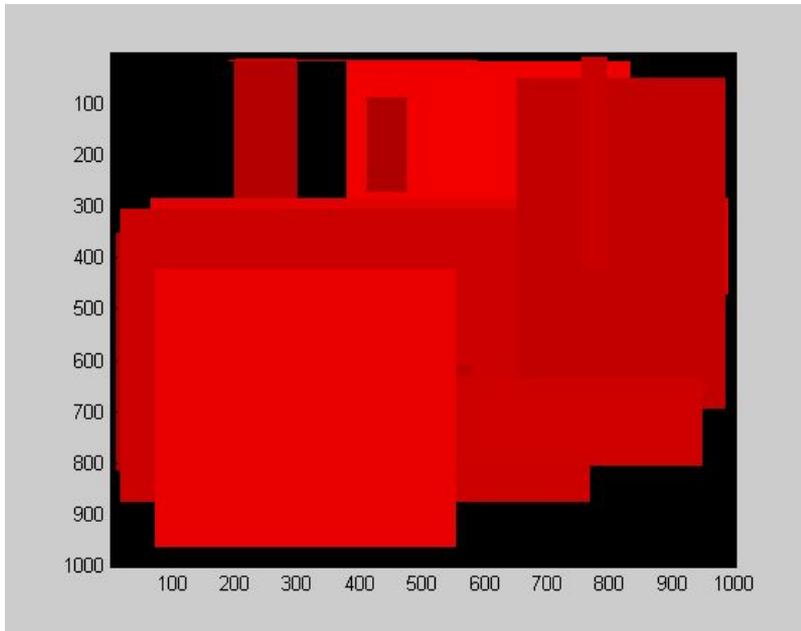


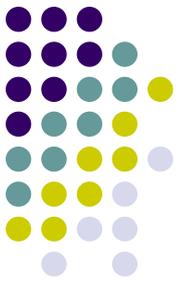
# Сегментация изображений

Обучения для 100 нейронов, размах – 1.  
2000 – примеров

Изображение

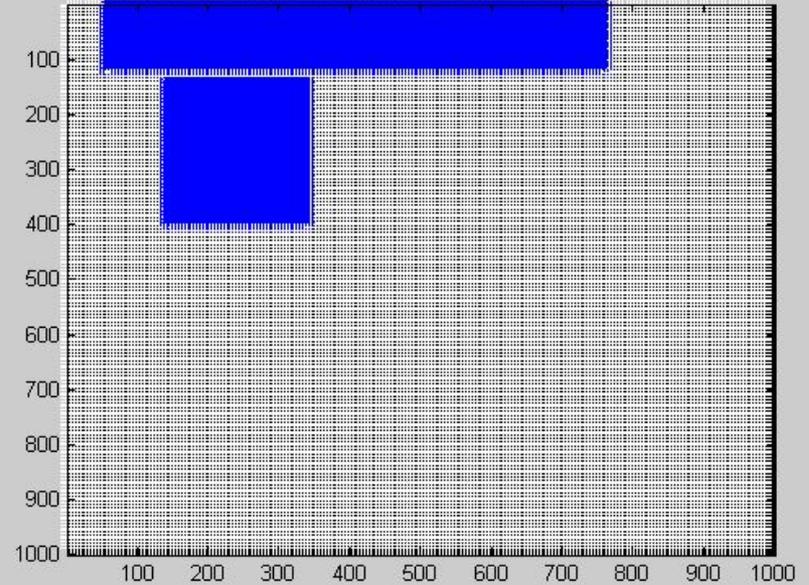
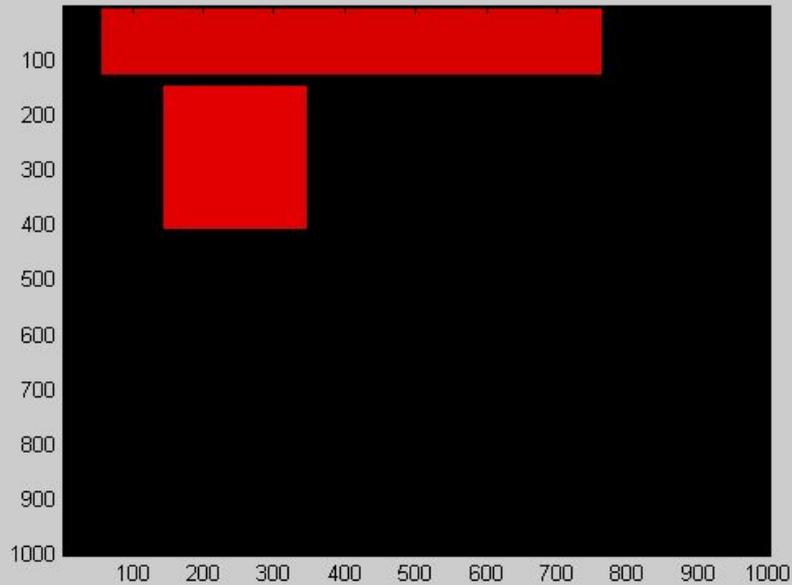
анализ

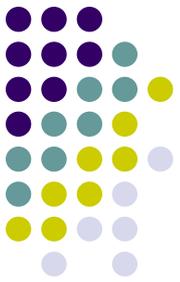




# Сегментация изображений

Результаты обучения:  
Тестовое изображение Анализ





# Литература

- 1. Комарцова Л.Г. Максимов А.В. Нейрокомпьютеры – М.:Из-во МГТУ - 2004.*
- 2. Уоссерман Ф. Нейрокомпьютерная техника: Теория и практика*