Кравцова Александра Николаевна, группа 12-1

Лабораторная работа № 2

Вариант № 8

Распознавание образов, описываемых гауссовскими случайными векторами с одинаковыми матрицами ковариаций

Цель работы

Синтезировать алгоритмы распознавания образов, описываемых гауссовскими случайными векторами с одинаковыми матрицами ковариаций. Исследовать синтезированные алгоритмы распознавания с точки зрения ожидаемых потерь и ошибок.

Задание

Реализовать алгоритм распознавания образов, описываемых гауссовскими случайными векторами с заданными параметрами. Получить матрицы ошибок на основе аналитических выражений и вычислительного эксперимента. Провести анализ полученных результатов и представить его в виде выводов по проделанной работе.

Предварительные данные:

$\boldsymbol{\mu}\mathbf{1}$=[2 -3], $\boldsymbol{\mu}\mathbf{2}$=[1 10], **C**=[4 -2; -2 4].

Код программы (Желтым выделены отличные от шаблона фрагменты):

```
%Файл pr52_rec_gaus_eq. Синтез и анализ алгоритмов распознавания ГСВ с одинаковой
%матрицей ковариации  (двумерный вектор признаков)
close all;

%1.Задание исходных данных
n=2; M=2; %%размерность признакового пространства  и число классов
K = 200; %количество статистических испытаний

m = [2 -3; 1 10]';    % координаты центров классов (2,-3) и (1,6)
% априорные вероятности классов (доля образов каждого класса в общей выборке)
pw = [0.5, 0.5];
np=sum(pw); pw=pw/np;

C = [4 -2; -2 4];    % матрица ковариаций классов
C_ = C^-1;
D = C(1,1);

% 1.1. Визуализация исходной совокупности образов
N = K * M;
NN = zeros(M, 1);
for k = 1 : M - 1
    NN(k) = uint16(N * pw(k));
end;
NN(M) = N - sum(NN);
label = {'bo', 'r+', 'k*', 'gx'};

IMS = [];   %общая совокупность образов
figure; hold on; title('Исходные метки образов');
for i=1:M,%цикл по классам
    ims = repmat(m(:,i), [1, NN(i)]) + randncor(n,NN(i),C); %генерация K образов i-го класса
    if (n == 2)
        plot(ims(1, :), ims(2, :), label{i}, 'MarkerSize', 10, 'LineWidth', 1);
    elseif (n == 3)
        plot3(ims(1, :), ims(2, :), ims(3, :), label{i}, 'MarkerSize', 10, 'LineWidth', 1);
```

```
    end;
    IMS = [IMS, ims];   %добавление в общую совокупность образов
end;

%2.Расчет разделяющих функций и матрицы вероятностей ошибок распознавания
G=zeros(M,n+1); PIJ=zeros(M); l0_=zeros(M);
for i = 1 : M,
    G(i,1:n)=(C_*m(:,i))';     G(i,n+1)=-0.5*m(:,i)'*C_*m(:,i);
    for j=i+1:M,
        l0_(i,j)=log(pw(j)/pw(i));
        h=0.5*(m(:,i)-m(:,j))'*C_*(m(:,i)-m(:,j)); sD=sqrt(2*h);
        PIJ(i,j)=normcdf(l0_(i,j),h,sD); PIJ(j,i)=1-normcdf(l0_(i,j),-h,sD);
    end;
    PIJ(i,i)=1-sum(PIJ(i,:));%нижняя граница вероятности правильного распознавания
end;

% 2.1. Визуализация результатов распознавания образов
figure; hold on; title('Результат классификации образов');
for i = 1 : N,%цикл по всем образам совокупности
    z = [IMS(:, i); 1]; %значение очердного образа из общей совокупности
    u=G*z+log(pw');%вычисление значения разделяющих функций
    [ui,iai]=max(u);%определение максимума (iai - индекс класса)
    if (n == 2)
        plot(IMS(1, i), IMS(2, i), label{iai}, 'MarkerSize', 10, 'LineWidth', 1);
    elseif (n == 3)
        plot3(IMS(1,   i),   IMS(2,   i),   IMS(3,   i),   label{iai},   'MarkerSize',   10,
'LineWidth', 1);
    end;
end;

%3.Тестирование алгоритма методом статистических испытаний
x=ones(n+1,1); Pc_=zeros(M);%экспериментальная матрица вероятностей ошибок
for k=1:K,%цикл по числу испытаний
    for i=1:M,%цикл по классам
        [x_,px]=randncor(n,1,C);
        x(1:n,1)=m(:,i)+x_;%генерация образа i-го класса
        u=G*x+log(pw');%вычисление значения разделяющих функций
        [ui,iai]=max(u);%определение максимума
        Pc_(i,iai)=Pc_(i,iai)+1;%фиксация результата распознавания
    end;
end;
Pc_=Pc_/K;
disp('Теоретическая матрица вероятностей ошибок');disp(PIJ);
disp('Экспериментальная матрица вероятностей ошибок');disp(Pc_);
%4.Визуализация областей принятия решений для двумерного случая
if n==2,
      xmin1=-4*sqrt(D)+min(m(1,:)); xmax1=4*sqrt(D)+max(m(1,:));
      xmin2=-4*sqrt(D)+min(m(2,:)); xmax2=4*sqrt(D)+max(m(2,:));
      x1=xmin1:0.05:xmax1; x2=xmin2:0.05:xmax2;
      axis([xmin1,xmax1,xmin2,xmax2]);%установка границ поля графика по осям
      figure; hold on; grid on;
      [X1,X2]=meshgrid(x1,x2); %матрицы значений координат случайного вектора
      x12=[X1(:),X2(:)];
      for i=1:M,
          f2=mvnpdf(x12,m(:,i)',C); %массив значений плотности распределения
          f3=reshape(f2,length(x2),length(x1));%матрица       значений       плотности
распределения
          [Ch,h]=contour(x1,x2,f3,[0.01,0.5*max(f3(:))],'Color','b','LineWidth',0.75);
clabel(Ch,h);
          for j=i+1:M,%изображение разделяющих границ
              wij=C_*(m(:,i)-m(:,j));
              wij0=-0.5*(m(:,i)+m(:,j))'*C_*(m(:,i)-m(:,j));
              f4=wij'*x12'+wij0;
              f5=reshape(f4,length(x2),length(x1));
              [Ch_,h_]                                                              =
contour(x1,x2,f5,[l0_(i,j)+0.0001],'Color','k','LineWidth',1.25);
          end;
    end;
    set(gca,'FontSize',13);
    title('Области локализации классов и разделяющие границы','FontName','Courier');
    xlabel('x1','FontName','Courier'); ylabel('x2','FontName','Courier');
    strv1=' pw='; strv2=num2str(pw,'% G');
    text(xmin1+1,xmax2-1,                                                 [strv1,strv2],
'HorizontalAlignment','left','BackgroundColor',...
```

```
        [.8 .8 .8],'FontSize',12); legend('wi','gij(x)=0');hold off;
end;
%% Функция для генерации совокупности гауссовских случайных векторов
function [x,m]=randncor(n,N,C)
%Функция для генерации совокупности гауссовских случайных векторов
%с нулевым математическим ожиданием и матрицей ковариации C
%n-исходный порядок матрицы ковариации C (размер n*n)
%N-количество генерируемых векторов
%формирование верхней треугольной матрицы разложения Холецкого
%с определением порядка матрицы и размерности генерируемых векторов
[A,r]=chol(C);
%определние размерности генерируемых векторов
if r==0, m=n; else m=r-1; end;
%генерация матрицы реализаций m*N гауссовских независимых случайных величин
u=randn(m,N);
%получение матрицы реализаций N гауссовских коррелированных векторов размерности m
x=A'*u;
end
```

Графическое отображение результатов:

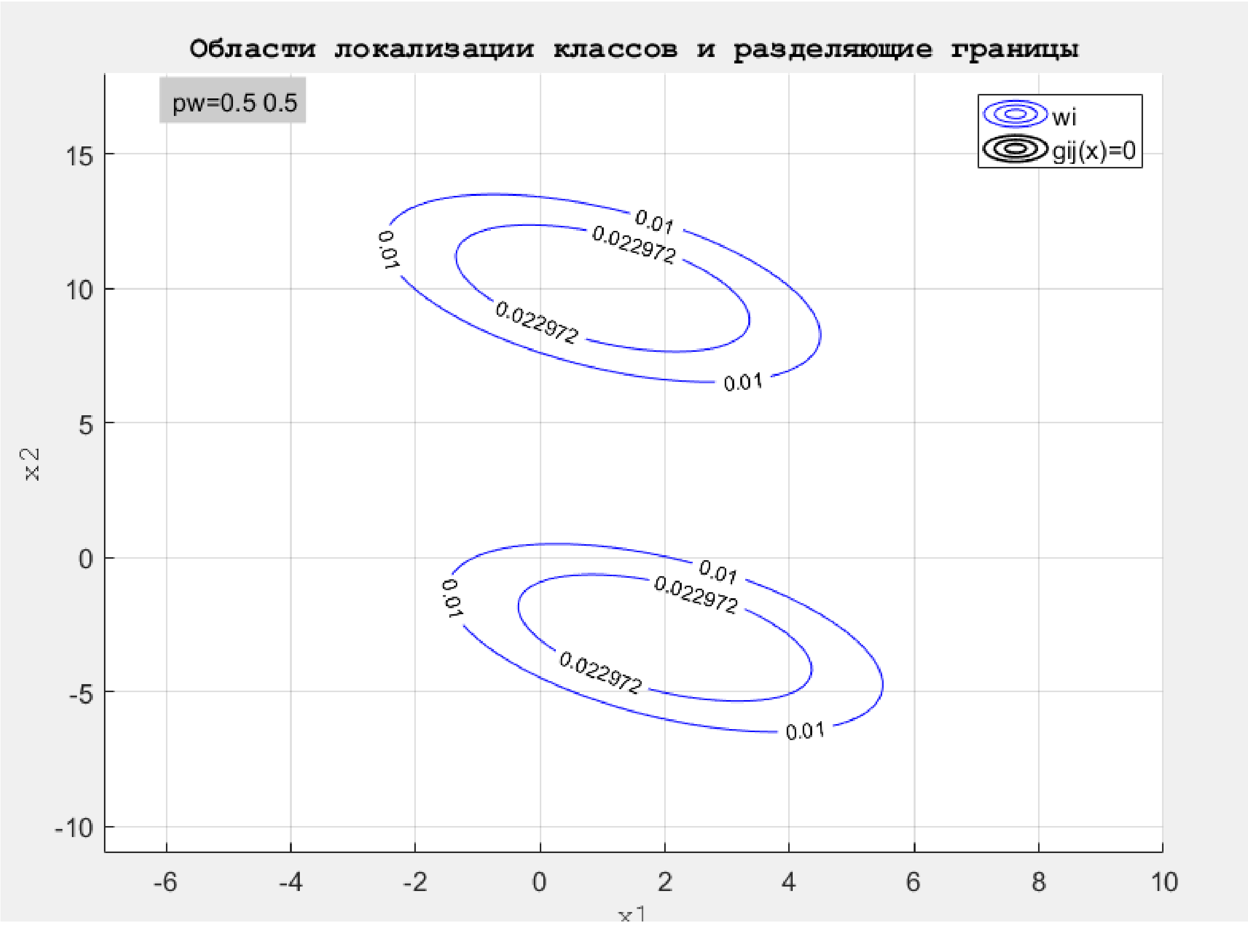


Рис.1

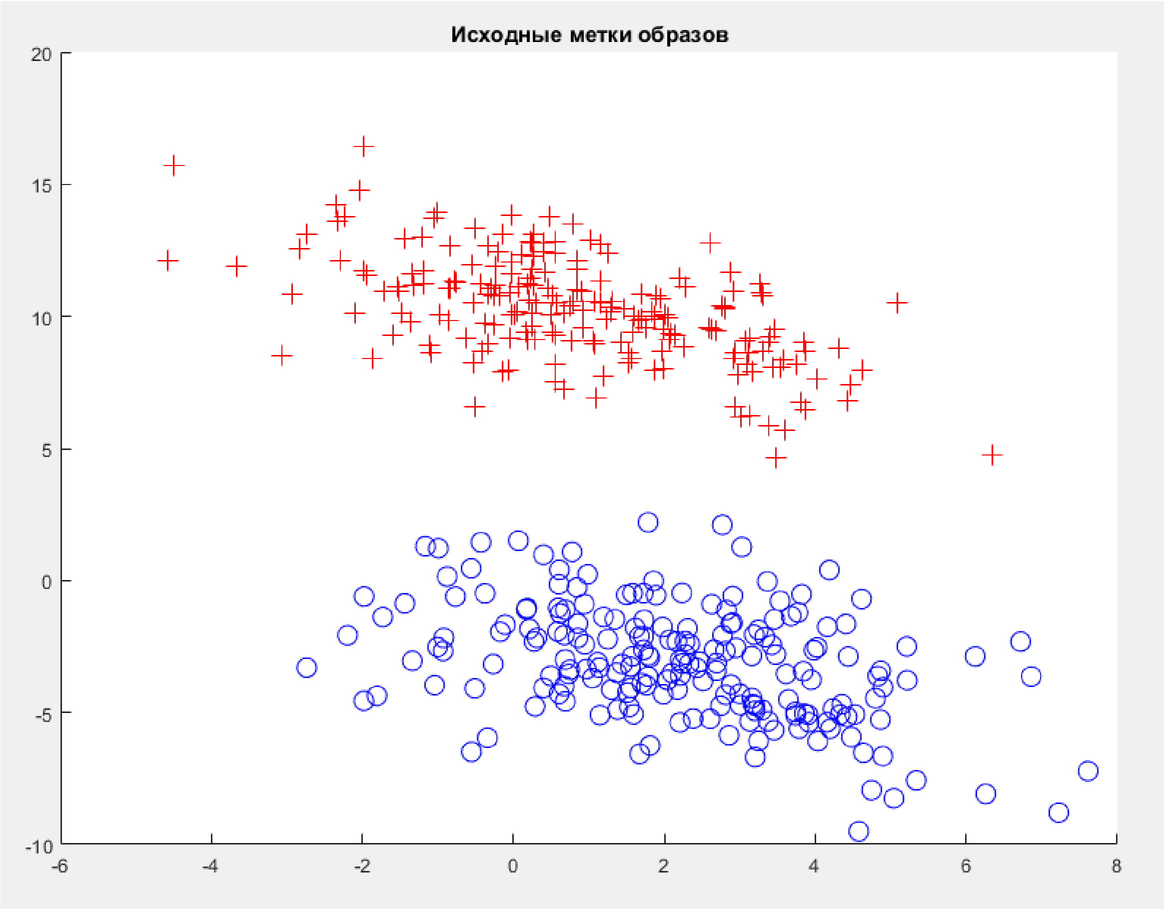

Исходные метки образов
20
15
10
5
0
-5
-10
-6
-4
-2
0
2
4
6
8

Рис.2

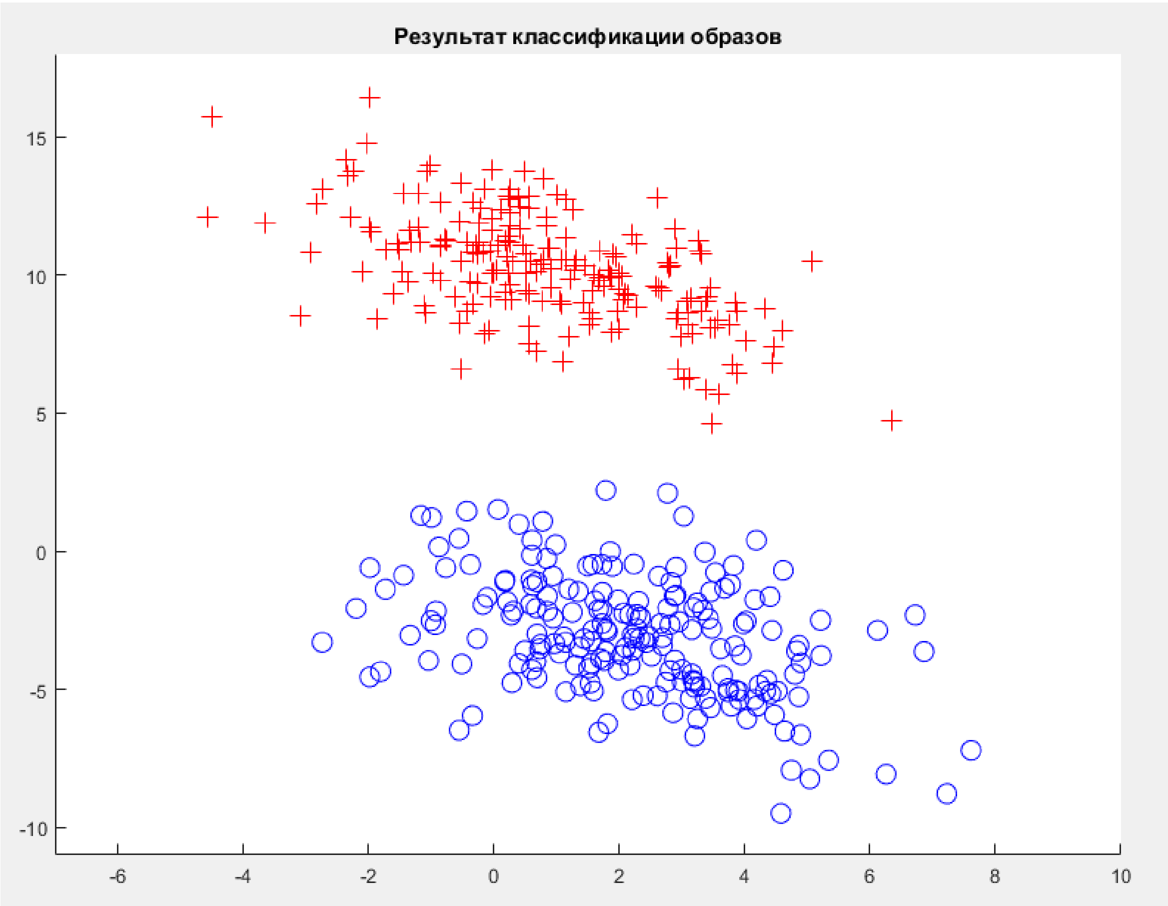

Результат классификации образов
15
10
5
0
-5
-10
-6
-4
-2
0
2
4
6
8
10

Рис.3

Результаты:

- **Полученные матрицы ошибок:**

Теоретическая матрица вероятностей ошибок

0.9999	0.0001
0.0001	0.9999

Экспериментальная матрица вероятностей ошибок

1	0
0	1

Дополнительные измерения:

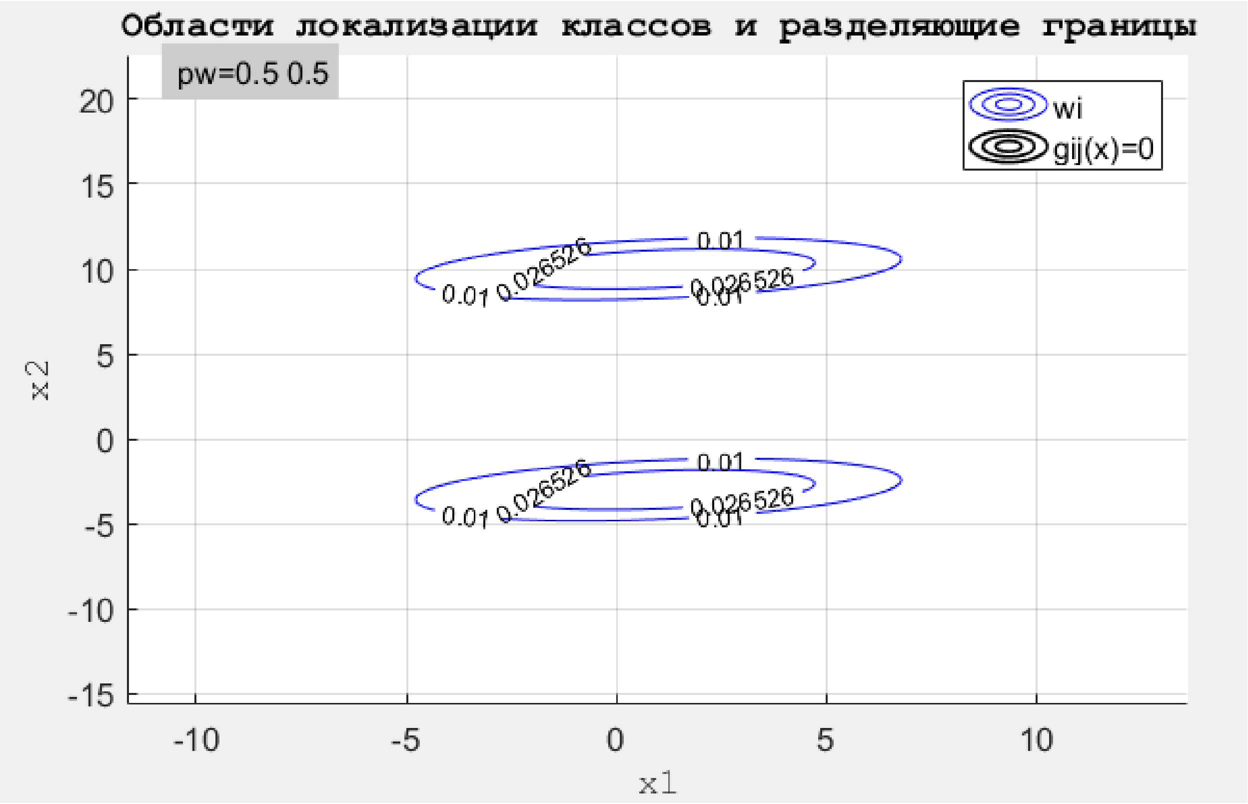
Измерение 1. Изменим исходные данные таким образом, чтобы увеличить протяженность области локализации образов всех классов (растянуть форму кластеров) в одном из направлений

- Исходные координаты: [2 -3; 1 10]
- Исходная матрица ковариации классов: [4 -2; -2 4]

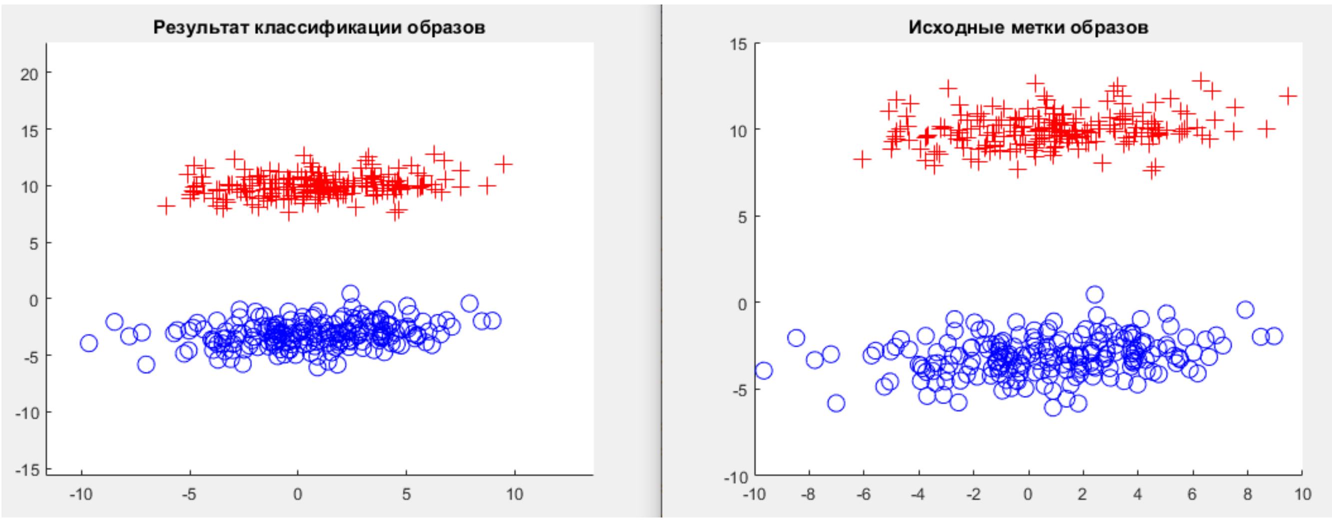
Изменим матрицу ковариации:

```
m = [1 -3; 1 10]
```

В результате получим:



Визуализация результата:



<u>Измерение 2.</u> Изменим исходные данные таким образом, чтобы в теоретической матрице ошибок увеличилась ошибка второго рода, а ошибка первого рода уменьшилась

Для начала дадим определение ошибкам первого и второго рода:

- Ошибки <u>первого</u> рода – при предъявлении образа одного класса, алгоритм классификации не относит образ к соответствующему классу (отвержение верной гипотезы).
- Ошибки <u>второго</u> рода – при предъявлении образа одного класса, алгоритм классификации относит образ к другому классу (принятие неверной гипотезы).

Данные до измерения:

- Исходные координаты: [2 -3; 1 10]
- Исходные априорные вероятности классов: [0.5, 0.5]
- Матрица вероятностей ошибок при исходных данных:

$$\begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix}$$

Измените исходные данные:

```
m = [8 3; 6 4]';     % координаты центров классов (2,-3) и (1,6)
pw = [0.8, 0.2];      % априорные вероятности классов
np=sum(pw); pw=pw/np;
```

В результате получим:

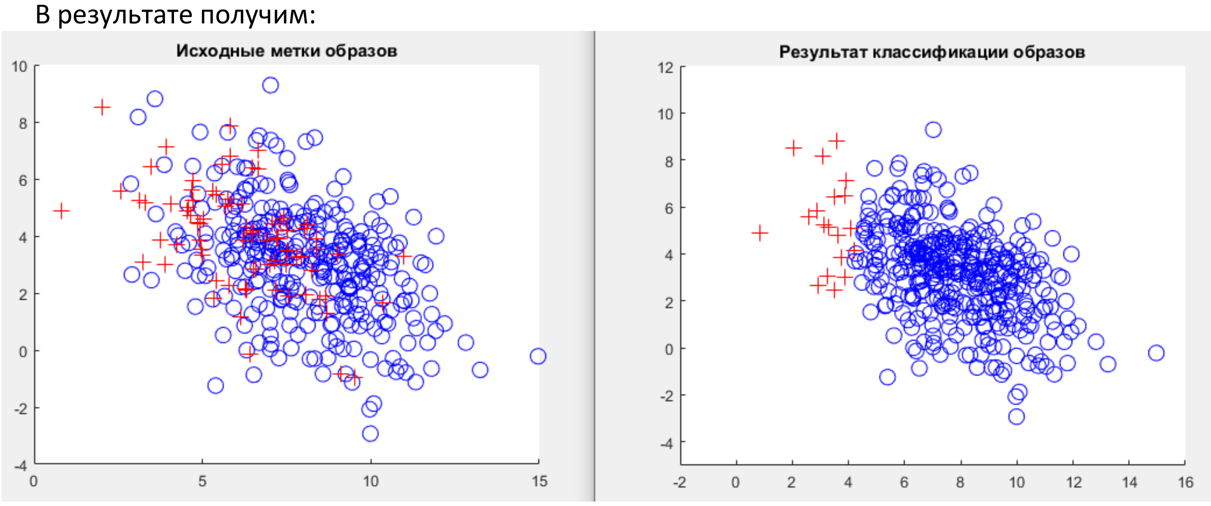


Рис.4

При этом матрица ошибок примет вид:

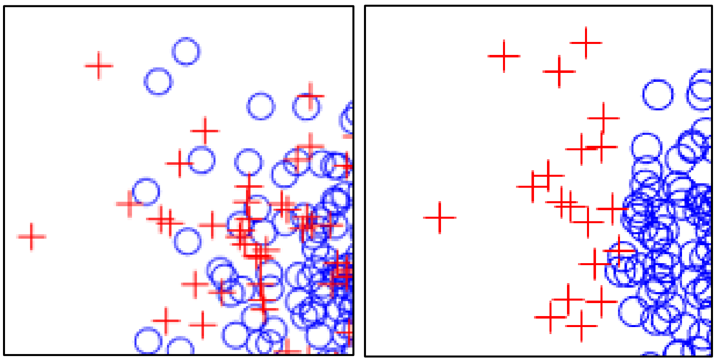
- Теоретическая матрица вероятностей ошибок

0.9704	0.0296
0.8123	0.1877

- Экспериментальная матрица вероятностей ошибок

0.9700	0.0300
0.8300	0.1700

Благодаря тому, что мы сместили центры классов «в кучу», а также уменьшили априорную вероятность второго класса (красный), ошибка второго рода увеличилась: на *рис.*4 наглядно видно, что второй класс потерял большую часть своих элементов, при этом «присвоил» элементы первого класса. Покажем наглядно, увеличив, и приглядевшись к *рис*.4:



Выводы:

В результате работы были синтезированы алгоритмы распознавания образов, описываемые гауссовскими случайными векторами с одинаковыми матрицами ковариаций. Также были исследованы синтезированные алгоритмы распознавания с точки зрения ожидаемых потерь и ошибок.