

Лабораторная работа № 2

Вариант № 4

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

$m_1=[0 \ -1]$, $m_2=[-4 \ 2]$, $m_3=[-1 \ 2]$, $C=[3 \ -2; -2 \ 3]$

Код программы (внесённые изменения в шаблон кода выделены)

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

```
%1.Задание исходных данных
```

```
n=2; M=3; % размерность признакового пространства и число классов
K = 1000; % количество статистических испытаний
```

```
m = [0 -1; -4 2; -1 2]; % мат. ожидания
```

```
% априорные вероятности классов (доля образов каждого класса в общей
выборке)
```

```
pw = [0.3, 0.3, 0.3];
```

```
np=sum(pw); pw=pw/np;
```

```
C = [3 -2; -2 3]; % матрица ковариаций классов
```

```
C_ = C^-1; % обратная ков. матрица
```

```
% 1.1. Визуализация исходной совокупности образов
```

```
% Определение числа образов в каждом классе, пропорционально pw
```

```
Ks = fix(pw .* K);
```

```
Ks(end) = K - sum(Ks(1 : end - 1));
```

```

label = {'bo', 'r+', 'k*', 'gx'}; % маркеры классов для визуализации
IMS = []; % общая совокупность образов (общая выборка)
figure; hold on; title('Исходные метки образов');
for i=1:M %цикл по классам
    ims = repmat(m(:,i), [1, Ks(i)]) + randnkor(n,Ks(i),C); %генерация Ks(i)
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 - теоретическая матрица ошибок
    PIJ(i,i)=1-sum(PIJ(i,:));%нижняя граница вероятности правильного
распознавания (на главной диагонали)
end

```

% 2.1. Визуализация результатов распознавания образов

```

figure; hold on; title('Результат классификации образов');
for i = 1 : K %цикл по всем образам совокупности
    z = [IMS(:, i); 1]; %значение очередного 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
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; % матрица ошибок, полученная экспериментально
```

```
% у нее такая же структура как и в PIJ, тока вычисляется численно, а не по  
% формулам
```

```
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
```

```
    D=1;
```

```
    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;
```

```
    figure; hold on; grid on;
```

```
    axis([xmin1,xmax1,xmin2,xmax2]); %установка границ поля графика по
```

```
осям
```

```
[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,[10_(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

```

Результаты выполнения задания

В результате работы программы были выведены вот такие графики:

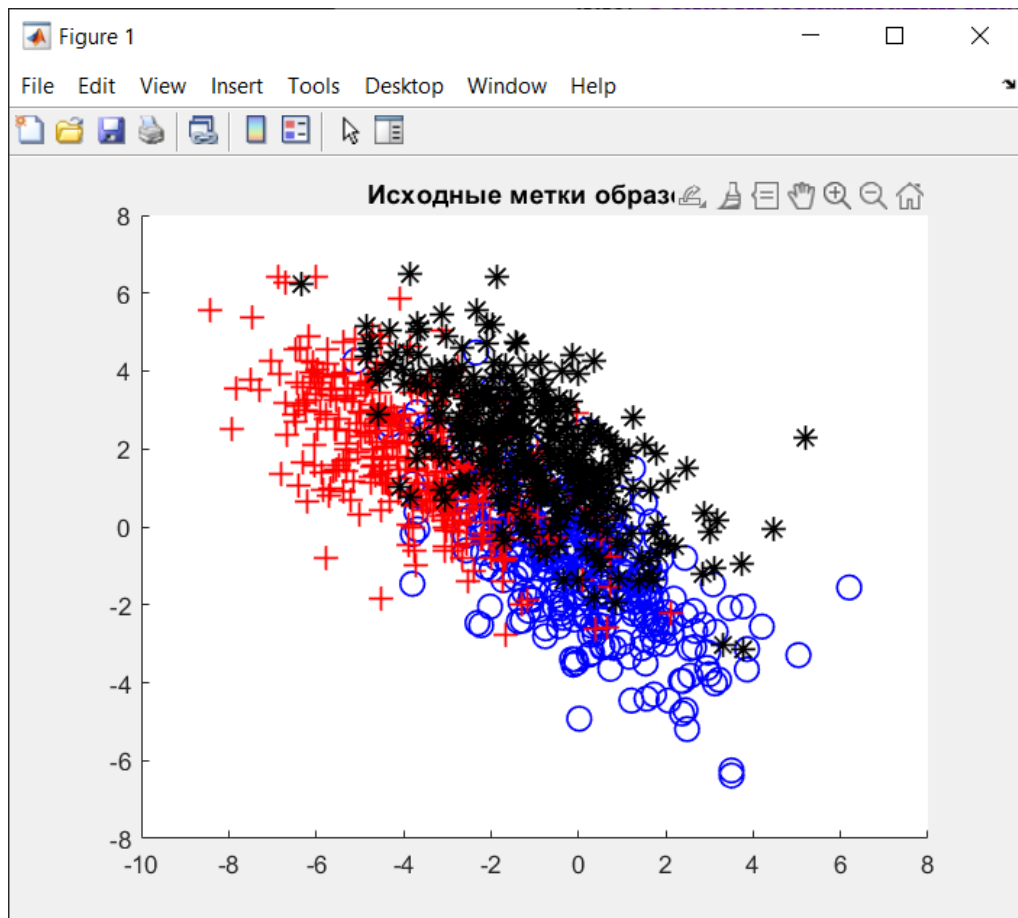


Рисунок 1 – Исходные метки образов

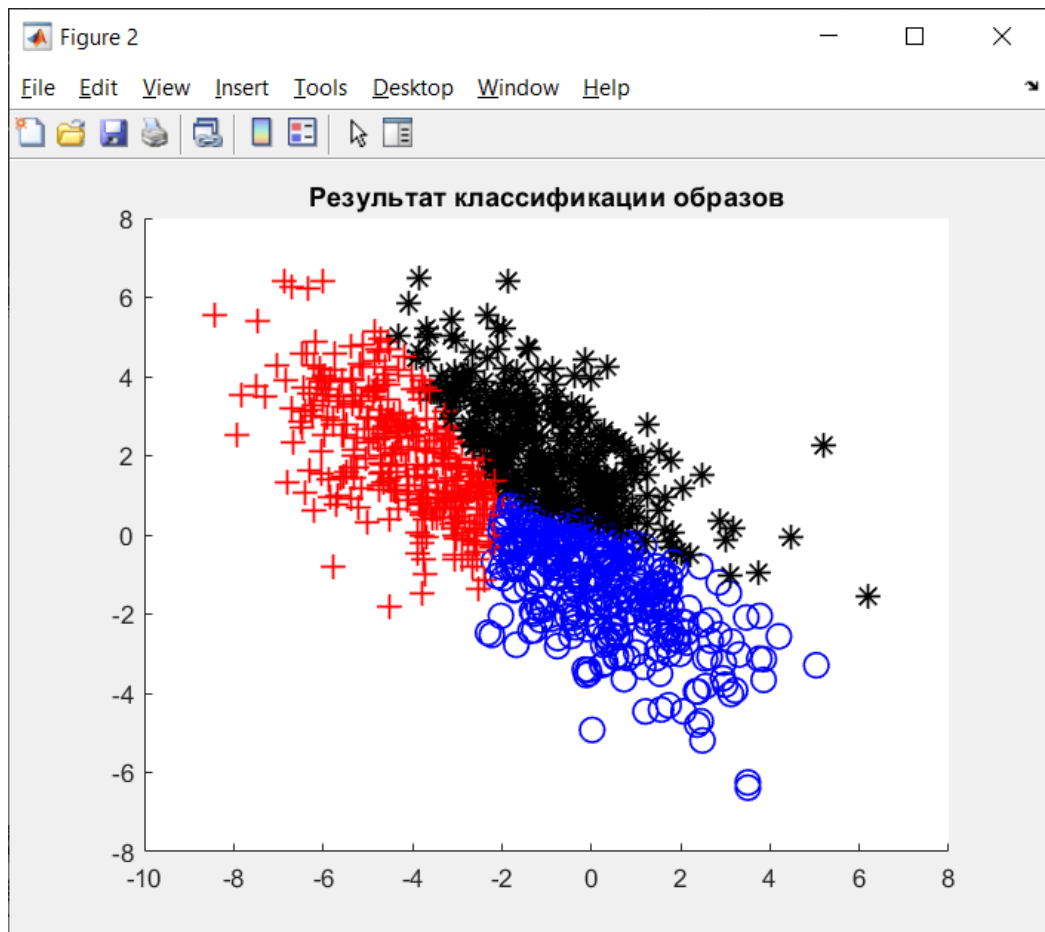


Рисунок 2 – Результат классификации образов

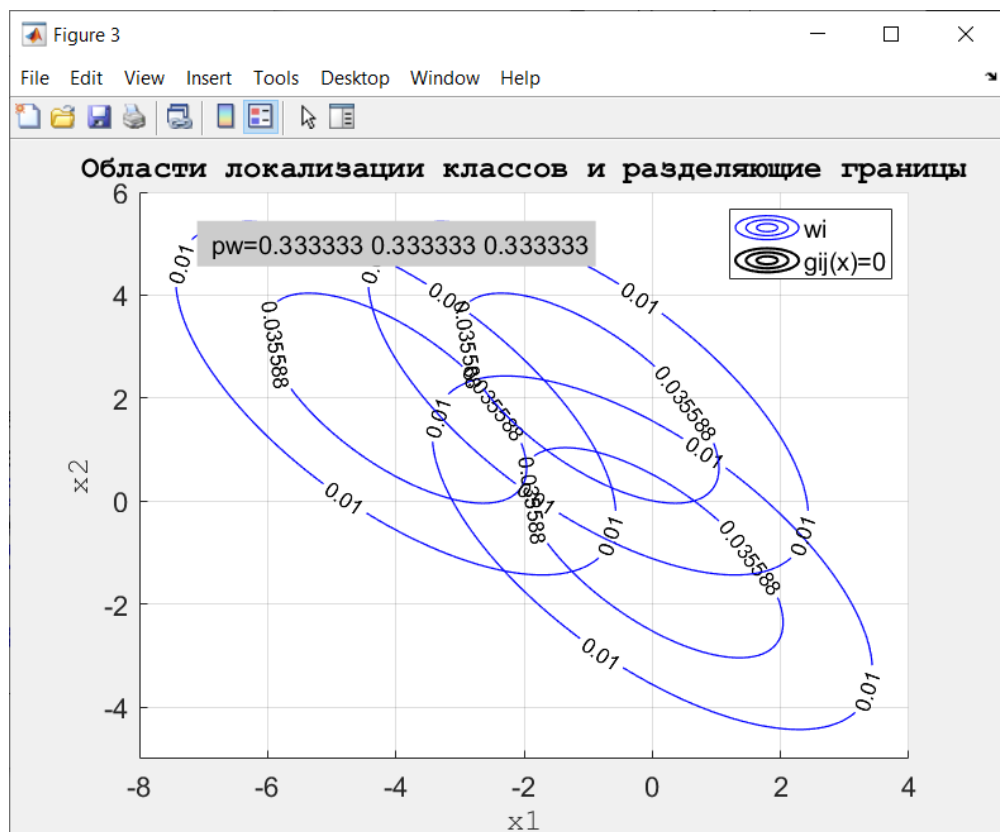


Рисунок 3 – Области локализации классов и разделяющие границы

```
Command Window

>> lab2_example
Теоретическая матрица вероятностей ошибок
    0.7060    0.1226    0.1714
    0.1226    0.7547    0.1226
    0.1714    0.1226    0.7060

Экспериментальная матрица вероятностей ошибок
    0.7550    0.1020    0.1430
    0.0960    0.8080    0.0960
    0.1410    0.0970    0.7620

fx >>
```

Рисунок 4 – Матрицы вероятностей ошибок (теоретическая и экспериментальная)

Дополнительные задания

а и б) Заменяем $pw = [0.3, 0.3, 0.3]$; на $pw = [0.4, 0.4, 0.2]$;

В результате увеличения априорных вероятностей pw , вероятность правильного распознавания увеличивается. Это видно на главной диагонали теоретической матрицы, т.к. значения на ней уменьшились.

```
>> lab2_example
Теоретическая матрица вероятностей ошибок
    0.6347    0.1939    0.1714
    0.0721    0.8052    0.1226
    0.1714    0.1226    0.7060

Экспериментальная матрица вероятностей ошибок
    0.8090    0.1110    0.0800
    0.1080    0.8430    0.0490
    0.2290    0.1350    0.6360

fx >>
```

Рисунок 5 – Матрицы вероятностей при $pw = [0.4, 0.4, 0.2]$

Увеличение априорных вероятностей также может привести к увеличению суммарной ошибки распознавания, поскольку классы с более высокой априорной вероятностью будут иметь больше веса в тестировании.

с) Чтобы увеличить ошибку первого рода и уменьшить ошибку второго рода, нужно сделать классы менее различимыми. Можно это сделать, например, увеличив разброс внутри классов. Для этого заменим в коде матрицу ковариаций классов с $C = \begin{bmatrix} 3 & -2 \\ -2 & 3 \end{bmatrix}$; на $C = \begin{bmatrix} 5 & -3 \\ -3 & 5 \end{bmatrix}$;

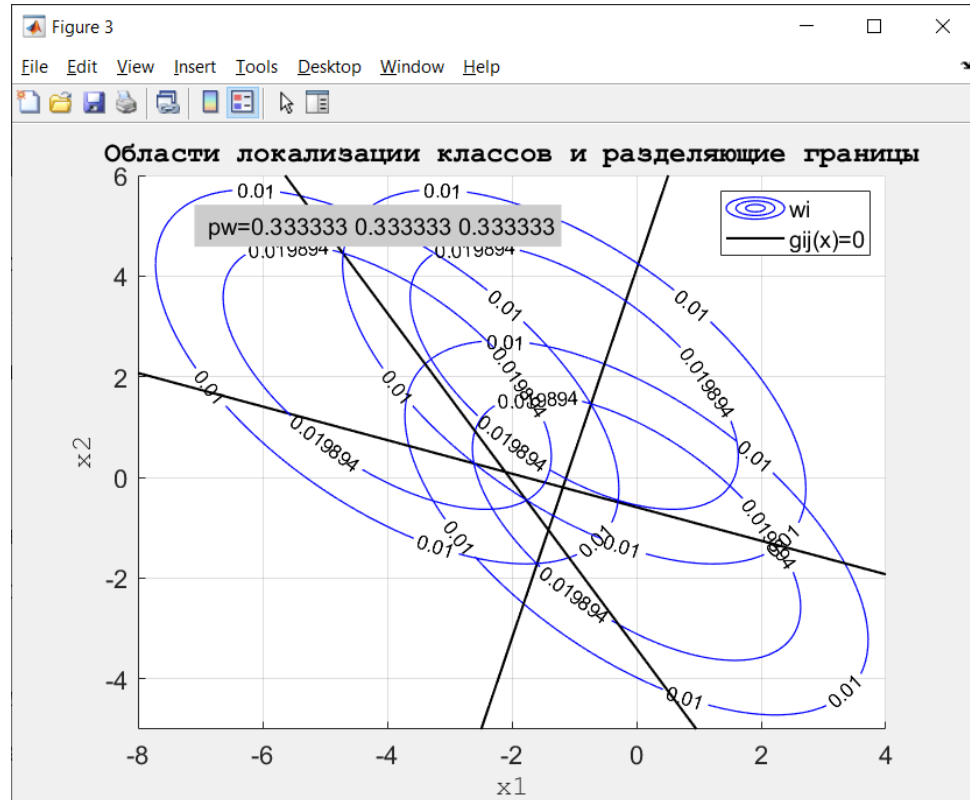


Рисунок 6 – Области классов при $C = \begin{bmatrix} 5 & -3 \\ -3 & 5 \end{bmatrix}$;

Области классов перекрываются больше, что означает, что точки из разных классов стали ближе друг к другу.

```
>> lab2_example
Теоретическая матрица вероятностей ошибок
    0.2875    0.2983    0.4141
    0.0984    0.5663    0.3353
    0.1156    0.1053    0.7791
```

Рисунок 7 – Теоретическая матрица вероятностей ошибок при $C = \begin{bmatrix} 5 & -3 \\ -3 & 5 \end{bmatrix}$;

Выводы

1) Элементы главной диагонали матрицы ошибок характеризуют вероятность правильного распознавания объектов каждого класса. Каждый

элемент (i, i) на главной диагонали показывает вероятность того, что объект класса i будет правильно распознан.

2) Элементы побочных диагоналей матрицы ошибок характеризуют вероятности ошибок классификации. Например, элемент (i, j) показывает вероятность ошибочной классификации объектов класса j как объектов класса i .

3) Формы кластеров объектов в пространстве используемых признаков определяются матрицей ковариаций. Если ковариации между признаками большие, кластеры будут вытянутыми и вытянутыми вдоль осей. Если ковариации малы, кластеры будут более круглыми и компактными. Таким образом, матрица ковариаций влияет на форму и ориентацию кластеров в пространстве признаков.