Беркман Даниела

М01ММ-23

**Отчет по 2 работе «Разделение множеств гиперплоскостями»**

Задача 1

Построить два набора случайных точек на плоскости, используя параметрическую формулу

где - заданные параметры,

- случайные величины,

равномерно распределена на интервале

равномерно распределена на интервале

В каждом наборе по 500 точек.

Для первого набора использовать параметры , для второго .

Рассчитать и изобразить на рисунке множества и прямую, разделяющую эти множества.

*Рассматривается задача разделения прямой разделяемых множеств*

clear

*Задание параметров для обоих наборов точек*

a1 = 3;

b1 = 1;

alpha1 = 0;

xc1 = -2;

yc1 = 1;

a2 = 4;

b2 = 1;

alpha2 = pi/4;

xc2 = 2;

yc2 = -1;

*Задание равномерно распределенных случайных величин на отрезках*

phi = unifrnd(0,pi\*2,500,1);

r = unifrnd(0,1,500,1);

x1 = [];

x2 = [];

y1 = [];

y2 = [];

*Построение двух наборов случайных точек на плоскости, используя параметрическую формулу*

for i = 1:500

x1 = [x1; xc1 + a1\*r(i).\*cos(phi(i)).\*cos(alpha1) - b1\*r(i).\*sin(phi(i)).\*sin(alpha1)];

y1 = [y1; yc1 + a1\*r(i).\*cos(phi(i)).\*sin(alpha1) + b1\*r(i).\*sin(phi(i)).\*cos(alpha1)];

x2 = [x2; xc2 + a2\*r(i).\*cos(phi(i)).\*cos(alpha2) - b2\*r(i).\*sin(phi(i)).\*sin(alpha2)];

y2 = [y2;yc2 + a2\*r(i).\*cos(phi(i)).\*sin(alpha2) + b2\*r(i).\*sin(phi(i)).\*cos(alpha2)];

end

A1 = [];

A2 = [];

B1 = [];

B2 = [];

*Задание задачи нахождения оптимальной прямой*

*(Линейное разделение множеств точек)*

for i = 1:500

A1 = [A1; [ 1, x1(i), y1(i)]];

A2 = [A2; [ 1, x2(i), y2(i)]];

*Ограничения*

B1 = [B1; 1];

B2 = [B2; -1];

end

lin = sdpvar(3,1);

F = [A1\*lin>=B1, A2\*lin<=B2];

optimize(F);

Lin = value(lin);

X1 = -5:0.1:5;

*Уравнение прямой:*

Y1 = -Lin(2)\*X1/Lin(3)-Lin(1)/Lin(3);

figure

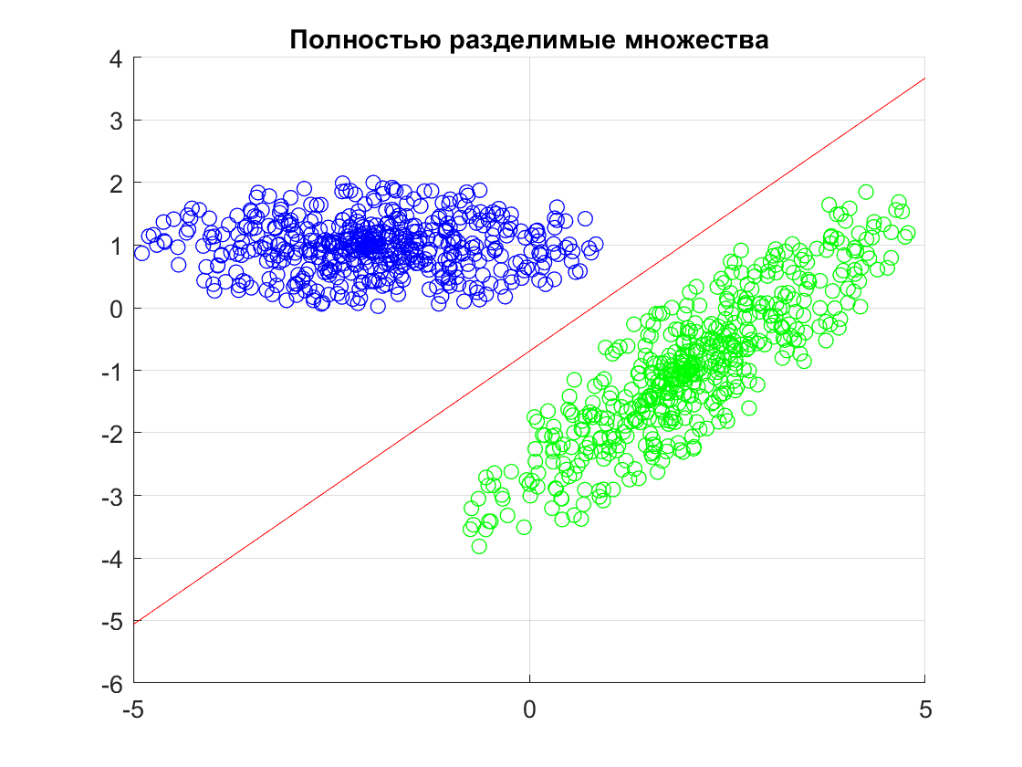
hold on

grid on

plot(x1,y1,'bo',x2,y2, 'go')

plot(X1,Y1,'red')

title("Полностью разделимые множества")



Задача 2

Используя те же формулы построить два множества, отличающиеся параметрами и .

Для первого множества взять , для второго .

Рассчитать и изобразить на рисунке множества и прямую, примерно разделяющую полученные множества.

*Рассматривается задача разделения прямой неразделяемых множеств*

clear

a12 = 3;

b12 = 2.5;

alpha12 = 0;

xc12 = -2;

yc12 = 1;

a22 = 4;

b22 = 2;

alpha22 = pi/4;

xc22 = 2;

yc22 = -1;

phi = unifrnd(0,pi\*2,500,1);

r = unifrnd(0,1,500,1);

x12 = [];

x22 = [];

y12 = [];

y22 = [];

for i = 1:500

x12 = [x12; xc12 + a12\*r(i).\*cos(phi(i)).\*cos(alpha12) - b12\*r(i).\*sin(phi(i)).\*sin(alpha12)];

y12 = [y12; yc12 + a12\*r(i).\*cos(phi(i)).\*sin(alpha12) + b12\*r(i).\*sin(phi(i)).\*cos(alpha12)];

x22 = [x22; xc22 + a22\*r(i).\*cos(phi(i)).\*cos(alpha22) - b22\*r(i).\*sin(phi(i)).\*sin(alpha22)];

y22 = [y22; yc22 + a22\*r(i).\*cos(phi(i)).\*sin(alpha22) + b22\*r(i).\*sin(phi(i)).\*cos(alpha22)];

end

A12 = [];

A22 = [];

B12 = [];

B22 = [];

lin2 = sdpvar(3,1);

for i = 1:500

A12 = [A12; [ 1, x12(i), y12(i)]];

A22 = [A22; [ 1, x22(i), y22(i)]];

B12 = [B12; 1];

B22 = [B22; -1];

end

*До этого момента все комментарии идентичны задаче 1*

*Задание задачи нахождения оптимальной прямой*

*(Приближенное линейное разделение неразделимых множеств)*

*Создание вектора переменных, состоящего из:*

lin2 = sdpvar(1003,1);

*В первых 3-х лежат параметры разделяющей прямой* ()

solve = lin2(1:3);

*Дальше лежат значения для u и v*

u = lin2(4:503);

v = lin2(504:1003);

*К ограничениям добавляется положительность векторов u и v*

F2 = [A12\*solve>=B12-u, A22\*solve<=B22+v, u>=0, v>=0];

*Целевой функцией будет*

h2 = sum(u)+sum(v);

optimize(F2,h2);

Lin2 = value(lin2);

X2 = -5:0.1:5;

Y2 = -Lin2(2)\*X2/Lin2(3)-Lin2(1)/Lin2(3);

Lin2(1);

Lin2(2);

Lin2(3);

figure

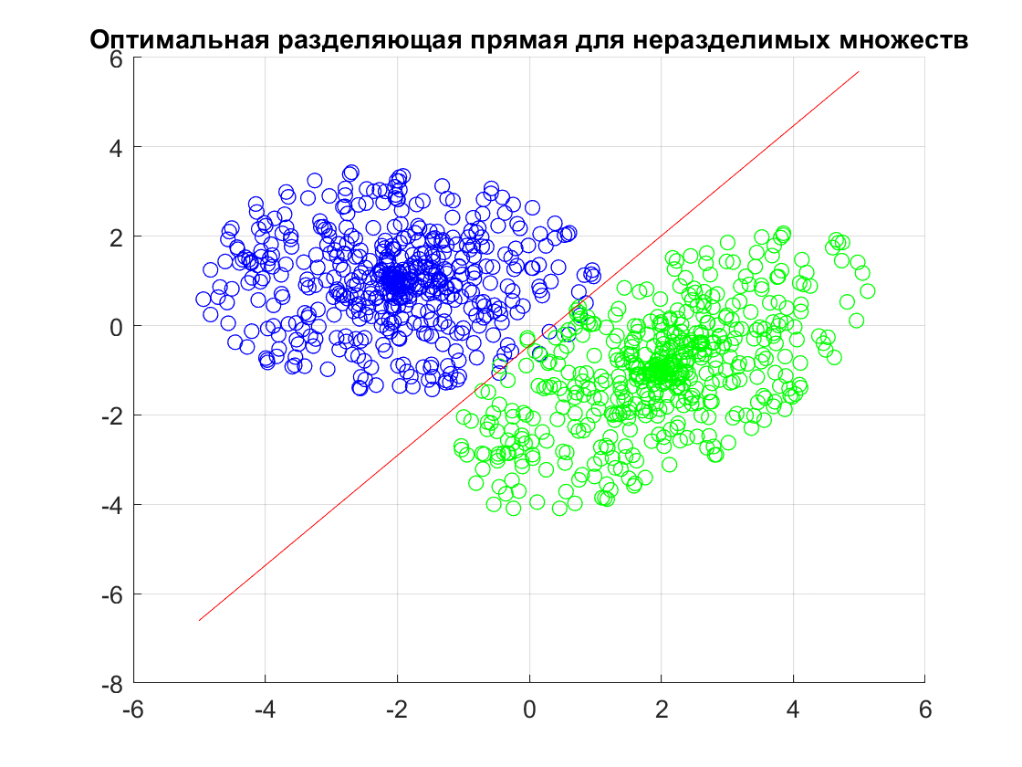
hold on

grid on

plot(x12,y12,'bo',x22,y22,'go')

plot(X2,Y2,'red')

title("Оптимальная разделяющая прямая для неразделимых множеств")



Задача 3

В базе MNIST (будет выслана в виде mat-файла) выбрать изображения из обучающей выборки, соответствующие двум различным цифрам (можно уменьшить количество изображений).

Построить несколько изображений символов из базы.

Построить гиперплоскость, примерно разделяющую множества изображений.

Проверить, как построенная гиперплоскость разделяет изображения из тестовой последовательности.

Найти процент ошибочных классификаций.

clear

*Загрузка mat-файла*

data = load('mnist.mat');

*Разделение на обучающую и тестовую выборку*

train = data.training;

test = data.test;

*Сначала рассмотрим обучающую выборку и решим для нее задачу оптимизации на 2х цифрах*

X\_train = train.images; % Обучающие изображения

y\_train = train.labels; % Метки классов для обучающего набора

im1 = [];

im2 = [];

num1 = 0;

num2 = 0;

*Выбираем 2 цифры (у меня 4, 0) и строим из квадратной картинки вектор, размером 784(=28\*28)*

for i = 1:60000

if y\_train(i) == 4

im1 = [im1, reshape(X\_train(:,:,i),784,1)];

num1 = num1+1;

end

if y\_train(i) == 0

im2 = [im2, reshape(X\_train(:,:,i),784,1)];

num2 = num2+1;

end

end

*Построение 3х примеров каждой цифры*

figure

for i = 1:3

subplot(2,3,i);

imshow(reshape(im1((i-1)\*784+1:i\*784), 28, 28));

title('Class 4');

subplot(2,3,i+3);

imshow(reshape(im2((i-1)\*784+1:i\*784), 28, 28));

title('Class 0');

end

*Задание задачи нахождения оптимальной разделяющей гиперплоскости в поле*

*(Классификация с помощью опорных векторов)*

A\_svm = sdpvar(784,1);

b\_svm = sdpvar(1);

u\_svm = sdpvar(1, num1);

v\_svm = sdpvar(1, num2);

F\_svm = [A\_svm'\*im1+b\_svm>=1-u\_svm, A\_svm'\*im2+b\_svm<=-1+v\_svm, u\_svm>=0, v\_svm>=0];

*Целевой функцией будет* *подбирается эмпически*

g = 10;

h\_svm = norm(A\_svm) + g\*(sum(u\_svm)+sum(v\_svm));

optimize(F\_svm, h\_svm, sdpsettings('solver', 'sdpt3','debug',1))

A\_SVM = value(A\_svm);

B\_SVM = value(b\_svm);

*Теперь проверяем на тестовых примерах как построилась оптимальная гиперплоскость*

X\_test = test.images; % Тестовые изображения

y\_test = test.labels; % Метки классов для тестового набора

im1\_test = [];

im2\_test = [];

num1\_test = 0;

num2\_test = 0;

for i = 1:10000

if y\_test(i) == 4

im1\_test = [im1\_test, reshape(X\_test(:,:,i),784,1)];

num1\_test = num1\_test+1;

end

if y\_test(i) == 0

im2\_test = [im2\_test, reshape(X\_test(:,:,i),784,1)];

num2\_test = num2\_test+1;

end

end

error1 = 0;

error2 = 0;

for i=1:num1\_test

if A\_SVM'\*im1\_test(:,i)+B\_SVM>=0

error1 = error1+1;

end

end

for i=1:num2\_test

if A\_SVM'\*im2\_test(:,i)+B\_SVM<=1

error2 = error2+1;

end

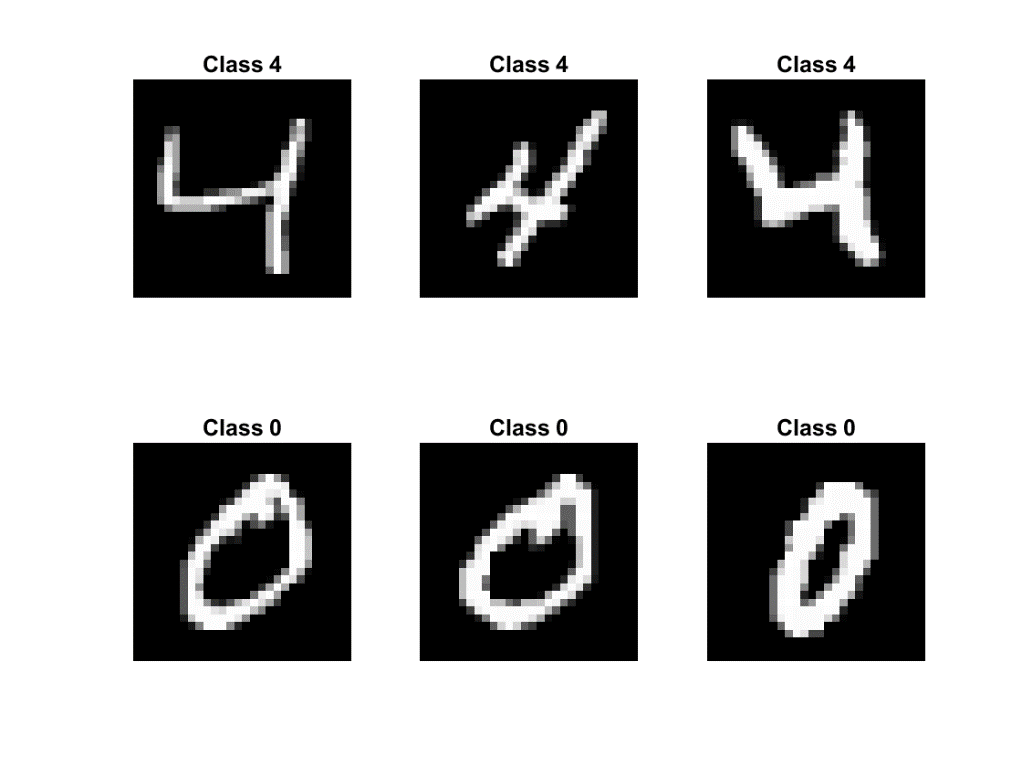
end

*Подсчет процента ошибочных классификаций*

err1 = 100-(error1/num1\_test)\*100

err2 = 100-(error2/num2\_test)\*100

*Примеры рассматриваемых цифр из набора MNIST:*



*Подсчет процента ошибочных классификаций:*

*SVM неправильно классифицировал 0.5092% 4 и 0.1020% 0, было произведено 39 итераций за 1 минуту 51 секунду.*