Путин Павел Александрович, группа 7-1

Лабораторная работа № 5

**Вариант № 4d**

Исследование непараметрических алгоритмов оценивания плотности распределения случайной величины

**Цель работы**

Исследовать алгоритмы оценивания плотности распределения случайных величин и случайных векторов на основе методов Парзена и k ближайших соседей.

**Задание**

Вычислить абсолютную ошибку оценивания плотности распределения случайного вектора в двумерном пространстве признаков при использовании оценки Парзена. Построить график зависимости ошибки оценивания от величины параметра прямоугольной оконной функции.

**Код программы (внесённые изменения в шаблон кода выделены)**

# Определение зависимости ошибки оценивания от величины параметра оконной функции

% Пример вар.4. Вычислить абсолютную ошибку оценивания плотности распределения

% случайного вектора в двумерном пространстве признаков при использовании оценки Парзена. Построить график зависимости ошибки оценивания от величины параметра оконной функции

clear all; close all;

%% Здесь только Двумерный случай

% ЗДЕСЬ задаются перебираемые занчения величины r на основе которой

% вычисляется параметр оконной функции

RR = 0.1 : 0.1 : 0.9;

err = RR \* 0; % массив значений ошибок заполненный нулями

% ЗДЕСЬ добавляется цикл по числу элементов RR

for tt = 1 : numel(RR)

% 1. Исходные данные

n = 2; % n - размерность вектора наблюдений

N = 2000; % количество используемых для оценки векторов

r = RR(tt); % ЗДЕСЬ подставляем очередное значение из массива RR

h\_N = N ^ (-r / n); % расчет параметра размера окна

kl\_kernel = 3; % ключ выбора ядра оценки (см. описание функции vkernel

% 2.Генерация отсчетов эталонной плотности (в виде смеси гауссиан) для двумерного случая

% Параметры распределения смеси гауссовских случайных векторов;

M = 3; % количество компонентов в смеси

ps = [0.2, 0.2, 0.6]; % вероятности появления СВ различных типов в смеси

% Расчет матрицы ковариаций ГСВ смеси

D = 0.2;

ro = -log(0.7); % дисперсия и коэффициент корреляции cоседних элементов

% Расположение математических ожиданий компонентов смеси

m1 = [0; 0];

m2 = [1; 0];

m3 = [0; 1];

m = [m1, m2, m3];

C = zeros(n, n);

% Ковариационная матрица компонентов смеси

for i = 1 : n

for j = 1 : n

C(i, j) = D \* exp(-ro \* abs(i - j));

end

end

x1 = -2 : 0.1 : 3;

x2 = -2 : 0.1 : 3; % области значений СВ, для которой визуализируется оценка

[X1, X2] = meshgrid(x1, x2);

x = [X1( : ) X2( : )]'; % матрицы Х и Y координат отсчётов

% Значения эталонной плотности

p = ps(1) \* mvnpdf(x', m1', C) + ps(2) \* mvnpdf(x', m2', C) + ps(3) \* mvnpdf(x', m3', C);

% 3. Обучающая выборка

XN = zeros(n, N);

% генерация обучающей выборки

for i = 1 : N

u = rand;

% индекс принадлежности к компоненте смеси

if u < ps(1)

t = 1;

elseif u < ps(1) + ps(2)

t = 2;

else

t = 3;

end

XN( : , i) = randncor(n, 1, C) + m( : , t);

end

% 4. Оценка плотности по Парзену

p\_ = vkernel(x, XN, h\_N, kl\_kernel); % оценка плотности

% ЗДЕСЬ фиксируем абсолютную ошибку

err(tt) = mean(abs(p( : ) - p\_( : )));

end

% ЗДЕСЬ вместо п.6,7. выводим зависимость ошибки от величины r

figure;

plot(RR, err); % то значение по горизонтали, где достигается минимум - и есть наилучшее значение r

# Определение вида оконной функции, обеспечивающего оптимальную оценку плотности распределения

% Пример вар.4. Вычислить абсолютную ошибку оценивания плотности распределения

% случайного вектора в двумерном пространстве признаков при использовании оценки Парзена.

% Построить график зависимости ошибки оценивания от величины параметра оконной функции

clear all;

close all;

%% Здесь только Двумерный случай

% ЗДЕСЬ задаются перебираемые занчения величины r на основе которой

% вычисляется параметр оконной функции

RR = 0.1 : 0.1 : 0.9;

err = RR \* 0; % массив значений ошибок заполненный нулями

types = [11 12 2 3 4];

t = tiledlayout(2, 3);

t.Padding = 'compact';

t.TileSpacing = 'compact';

for kernel\_type = types

plot\_title = "";

% ЗДЕСЬ добавляется цикл по числу элементов RR

for tt = 1 : numel(RR)

% 1. Исходные данные

n = 2; % n - размерность вектора наблюдений

N = 2000; % количество используемых для оценки векторов

r = RR(tt); % ЗДЕСЬ подставляем очередное значение из массива RR

h\_N = N ^ (-r / n); % расчет параметра размера окна

kl\_kernel = kernel\_type; % ключ выбора ядра оценки (см. описание функции vkernel) !! 12 -> 3

% 2.Генерация отсчетов эталонной плотности (в виде смеси гауссиан) для двумерного случая

% Параметры распределения смеси гауссовских случайных векторов;

M = 3; % количество компонентов в смеси

ps = [0.2, 0.2, 0.6]; % вероятности появления СВ различных типов в смеси

% Расчет матрицы ковариаций ГСВ смеси

D = 0.2;

ro = -log(0.7); % дисперсия и коэффициент корреляции cоседних элементов

% Расположение математических ожиданий компонентов смеси

m1 = [0; 0];

m2 = [1; 0];

m3 = [0; 1];

m = [m1, m2, m3];

C = zeros(n, n);

% Ковариационная матрица компонентов смеси

for i = 1 : n

for j = 1 : n

C(i, j) = D \* exp(-ro \* abs(i - j));

end

end

x1 = -2 : 0.1 : 3;

x2 = -2 : 0.1 : 3; % области значений СВ, для которой визуализируется оценка

[X1, X2] = meshgrid(x1, x2);

x = [X1( : ) X2( : )]'; % матрицы Х и Y координат отсчётов

% Значения эталонной плотности

p = ps(1) \* mvnpdf(x', m1', C) + ps(2) \* mvnpdf(x', m2', C) + ps(3) \* mvnpdf(x', m3', C);

% 3. Обучающая выборка

XN = zeros(n, N);

% генерация обучающей выборки

for i = 1 : N

u = rand;

% индекс принадлежности к компоненте смеси

if u < ps(1)

t = 1;

elseif u < ps(1) + ps(2)

t = 2;

else

t = 3;

end

XN( : , i) = randncor(n, 1, C) + m( : , t);

end

% 4. Оценка плотности по Парзену

p\_ = vkernel(x, XN, h\_N, kl\_kernel); % оценка плотности

% ЗДЕСЬ фиксируем абсолютную ошибку

err(tt) = mean(abs(p( : ) - p\_( : )));

end

% ЗДЕСЬ вместо п.6,7. выводим зависимость ошибки от величины r

ax = nexttile;

plot(ax, RR, err); % то значение по горизонтали, где достигается минимум - и есть наилучшее значение r

hold on;

[ymin, imin] = min(err);

xmin = RR(imin);

plot(ax, xmin, ymin, 'ro');

% Text with coordinates of minimum

offset = .05; % vertical offset as a fraction of y-axis span. Change as needed.

text(xmin, ymin + diff(ylim) \* offset, ['(' num2str(xmin) ',' num2str(ymin) ')'])

% Enlarge y axis so that text is properly seen, if offset is negative

ylim(ylim + [diff(ylim) \* offset \* (offset < 0) 0])

hold off;

switch kl\_kernel

case 11

plot\_title = "Гауссовская функция c использованием диагональной матрицы";

case 12

plot\_title = "Гауссовская функция c использованием матрицы ковариации";

case 2

plot\_title = "Показательная функция";

case 3

plot\_title = "Оконная прямоугольная функция";

case 4

plot\_title = "Оконная треугольная функция";

end

title(ax, plot\_title);

end

**Результаты выполнения задания**

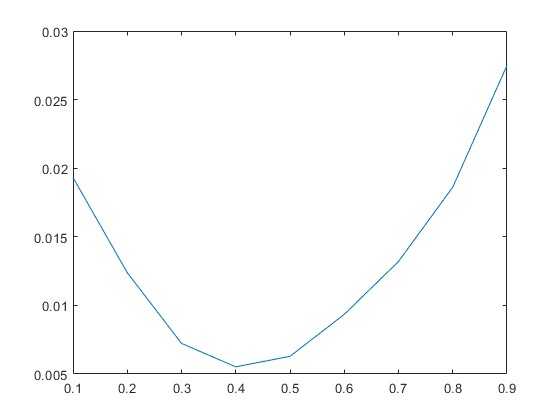
****

Рисунок - График зависимости ошибки оценивания от величины параметра прямоугольной оконной функции

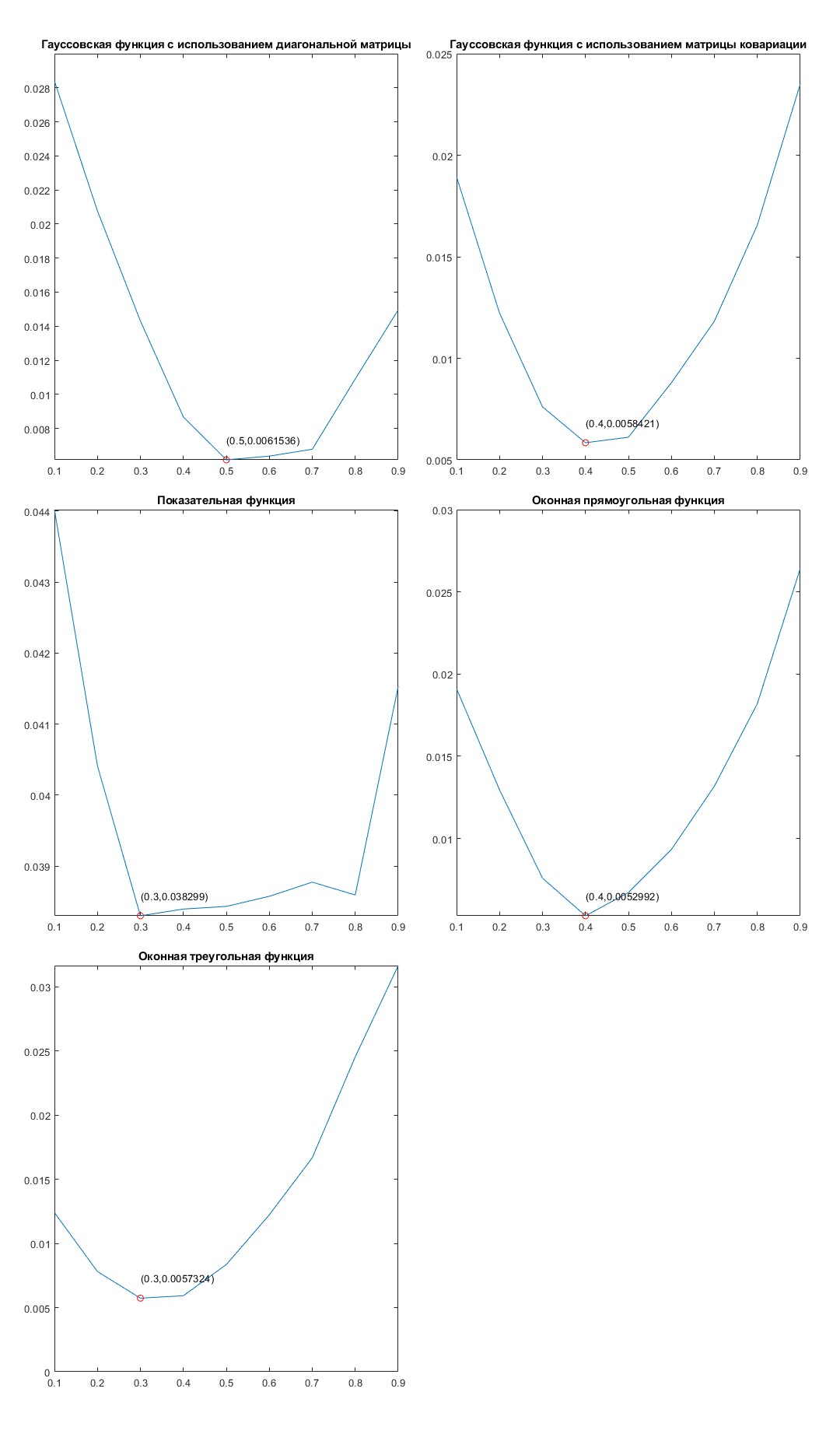


Рисунок - оценка плотности распределения при разных оконных функциях

# Выводы

1. Из графика на рисунке Рисунок 1 можно сделать вывод, что минимум ошибки оценивания по критерию Парзена достигается при значении параметра прямоугольной оконной функции равном 0,4.
2. Из графиков на рисунке Рисунок 2 можно сделать вывод, что оптимальную оценку плотности распределения (с наименьшей ошибкой по критерию Парзена) обеспечивает оконная треугольная функция с параметром 0,3.