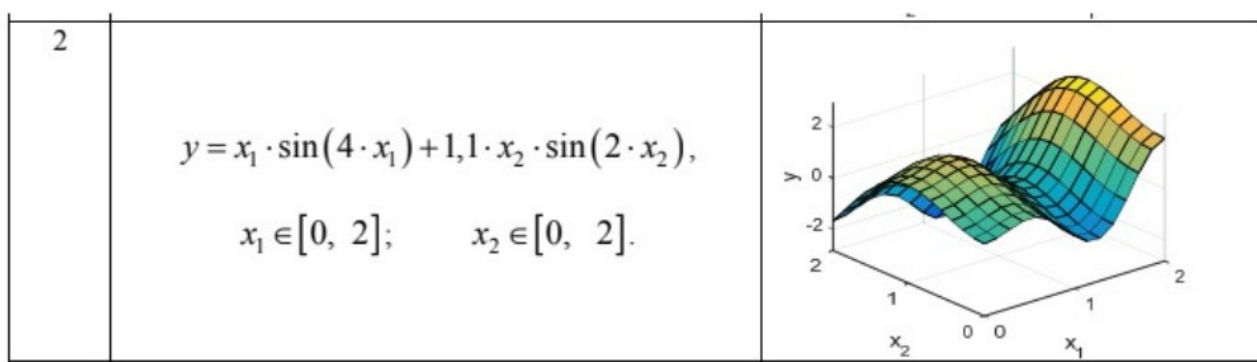


1. Борискин М.А.
2. 3341506/90401
3. 29.11.2020
4. Адаптация системы нечёткого вывода Мамдани с использованием средств нелинейной оптимизации пакета Optimization Toolbox.
5. Изучение особенностей построения адаптивных нечётких систем с использованием средств нелинейной оптимизации пакета Optimization Toolbox системы MatLab.
- 6.

Вариант задания: 2:



m-программа адаптации системы нечёткого вывода к нелинейной зависимости согласно варианту задания:

```
% Нечёткая аппроксимация зависимости: y=x1^2*sin(x2-1)
%
(LR_6_AdaptFIS.m)
%
% <<<          Адаптация системы нечёткого вывода Мамдани          >>>
% <<< с использованием функции нелинейной оптимизации fmincon >>>
% <<<          пакета Optimization Toolbox          >>>
%=====
=====
clear all          % очистка памяти (leaving the workspace empty)
clc               % очистка командного окна (Clear Command Window)
%-----
-----
% Построение графика функции y=x1^2*sin(x2-1)
% в области      x1 = [0, 2];  x2 = [0, 2]
%-----

n = 15;           % количество точек дискретизации
x1 = linspace(0, 2, n);
x2 = linspace(0, 2, n);
```

```

y = zeros(n, n);
for i = 1:n
    y(i,:) = x1.*sin(4*x1)+1.1*x2(i)*sin(2*x2(i));
end

%-----
h1 = figure(1);
set(h1,'Position',[622 541 524 407])
% get(h1,'Position')
surf(x1, x2, y)
axis([min(x1) max(x1) ...
      min(x2) max(x2) ...
      min(min(y)) max(max(y)) ]);
view(-40,30)
xlabel('x_1'); ylabel('x_2'); zlabel('y');
title('Исходная зависимость')

%-----
% Построение графика нечёткого отображения Мамдани
% с гауссовыми функциями принадл. термов вх.перем.
%-----

fisMg = readfis('2varMg'); % загрузка исходной нечёткой сист. с
диска
% fuzzy(fisMg) % вызов FIS-Editor с исходной
неч.сист.

yMg = zeros(n, n);
for i = 1:n
    yMg(i,:) = evalfis([x1; ones(size(x1))*x2(i)], fisMg)';
end

%-----

h2 = figure(2);
set(h2,'Position',[1154 541 524 407])
colormap('default')
surf(x1, x2, yMg)
axis([min(x1) max(x1) ...
      min(x2) max(x2) ...
      min(min(y)) max(max(y)) ]);
view(-40,30)
xlabel('x_1'); ylabel('x_2'); zlabel('y');
title('Система нечёткого вывода Мамдани до адаптации')

%-----
% Формирование обучающего и тестового массивов данных

```

```

%-----

Nob = 60;                                % количество точек обучающего
массива
Nts = 60;                                % количество точек тестового
массива

RndStt = 3;                              %
уст.начальн.сост.генер.случ.чисел
rand('state',RndStt);

% Входы:
xlob = min(x1)+(max(x1)-min(x1))*rand(Nob,1);
x2ob = min(x2)+(max(x2)-min(x2))*rand(Nob,1);
x1ts = min(x1)+(max(x1)-min(x1))*rand(Nts,1);
x2ts = min(x2)+(max(x2)-min(x2))*rand(Nts,1);

% Выходы:

yob = zeros(size(xlob));
for i = 1:Nob
    yob(i) = xlob(i)^2 * sin(x2ob(i)-1);
end

yts = zeros(size(x1ts));
for i = 1:Nts
    yts(i) = x1ts(i)^2 * sin(x2ts(i)-1);
end

%-----

h3 = figure(3);
set(h3,'Position',[91    541    524    407])
clf
hold on
plot(xlob, x2ob, 'bs', 'MarkerSize',4)
plot(x1ts, x2ts, 'ro', 'MarkerSize',4)
hold off
axis([min(x1)-0.01 max(x1)+0.01 ...
      min(x2)-0.01 max(x2)+0.01]);
xlabel('x1'); ylabel('x2');
title('Распределение данных обучающей (bs) и тестовой (or)
выборок')

%-----

pause(2);                                % пауза 2,0 с

figure(2);
colormap('white')                        %

```

```

hold on
plot3(xlob, x2ob, yob, 'bs', 'MarkerSize',3,
'MarkerFaceColor','b')
plot3(xlts, x2ts, yts, 'ro', 'MarkerSize',3,
'MarkerFaceColor','r')
hold off

%-----
% Процесс адаптации системы нечёткого вывода
%-----
% НАСТРАИВАЕМЫЕ ПАРАМЕТРЫ (19 шт.):
%   - весовые коэффициенты правил 5, 6, 7   (3 шт.);
%   - коэф. конц.термов переменных x1, x2, y (3+3+5=11 шт.);
%   - коорд.максимумв термов "средний" переменных x1, x2 (2 шт.);
%   - коорд.макс.терм."ниже сред.", "сред.", "выше сред." перемен.
%   y (3 шт.)
%-----
% Весовые коэффициенты правил
w0 = [1 1 1 1 1 1 1 1]-0.001; % начальное приближение
wL = [0 0 0 0 0 0 0 0]; % нижняя граница
wU = [1 1 1 1 1 1 1 1]; % верхняя граница

% Коэффициенты концентрации термов переменных
x1_s10 = fisMg.input(1).mf(1).params(1); % исх. знач. для x1
"низкий"
x1_s20 = fisMg.input(1).mf(2).params(1); % -//-
"средний"
x1_s30 = fisMg.input(1).mf(3).params(1); % -//-
"высокий"

x2_s10 = fisMg.input(2).mf(1).params(1); % исх. знач. для x2
"низкий"
x2_s20 = fisMg.input(2).mf(2).params(1); % -//-
"средний"
x2_s30 = fisMg.input(2).mf(3).params(1); % -//-
"высокий"

y_s10 = fisMg.output(1).mf(1).params(1); % исх.знач.для y "низкий"
y_s20 = fisMg.output(1).mf(2).params(1); % -//- "ниже
среднего"
y_s30 = fisMg.output(1).mf(3).params(1); % -//- "средний"

y_s40 = fisMg.output(1).mf(4).params(1); % -//- "выше
среднего"
y_s50 = fisMg.output(1).mf(5).params(1); % -//- "высокий"

x12ys0 = [x1_s10 x1_s20 x1_s30 ... % объедин.цент.конц. в
один вектор
x2_s10 x2_s20 x2_s30 ...

```

```

        y_s10    y_s20    y_s30    y_s40    y_s50];
x12ysL = 0.7*x12ys0;      % нижняя граница = начальное приближение -
30%
x12ysU = 1.3*x12ys0;      % верхняя граница = начальное приближение
+ 30%

% коорд.максимумв некотор.термов входных и выходных переменных
x1_c20 = fisMg.input(1).mf(2).params(2); % исх. знач. для x1
"средний"
x2_c20 = fisMg.input(2).mf(2).params(2); %      -//-      x2
"средний"
y_c20 = fisMg.output(1).mf(2).params(2); %      -//-      y "ниже
среднего"
y_c30 = fisMg.output(1).mf(3).params(2); %      -//-      y
"средний"
y_c40 = fisMg.output(1).mf(4).params(2); %      -//-      y "выше
среднего"

x12yc0 = [x1_c20  x2_c20  y_c20  y_c30  y_c40];

dx1 = 0.3*(max(x1)-min(x1));
dx2 = 0.3*(max(x2)-min(x2));
dy = 0.2*(max(max(y))-min(min(y)));
x12ycL = x12yc0 - [dx1 dx2 dy dy dy];      % нижняя граница
x12ycU = x12yc0 + [dx1 dx2 dy dy dy];      % верхняя граница

% Объединение настраиваемых параметров в один вектор
ParamFis0 = [w0  x12ys0  x12yc0];          % начальное приближение
ParamFisL = [wL  x12ysL  x12ycL];          % нижняя граница
ParamFisU = [wU  x12ysU  x12ycU];          % верхняя граница

% Масштабир.настраив.параметров (демасштаб. в функц.
F_changeFISmg)
Msht = [1  1  1  1  1  1  1  1  1  1  1  1  1  1  0.1  0.1  0.1
0.1  0.1  1  1  0.04  0.04  0.04];
%Msht = [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1]; %
отсутствие масшт.парам.
ParamFis0 = ParamFis0 .* Msht;
ParamFisL = ParamFisL .* Msht;
ParamFisU = ParamFisU .* Msht;

%-----
% ПАРАМЕТРЫ ОПТИМИЗАЦИИ

% optimset('fmincon')      % вывод структуры с инф.о парам.
алгоритма оптим.

% options = [];
options = optimset('Display', 'iter');      % вывод инф. на каждой
итерации

```

```

options.MaxIter = 25; % максимальное число
итераций
options.DiffMinChange = 0.0001;
options.DiffMaxChange = 0.2;
options.LargeScale = 'off';

%-----
% ОПТИМИЗАЦИЯ

[ParamFis_opt, sqrtFis, flag] = fmincon(@F_errFISmg, ParamFis0, [],
[], [], [], [], ...
    ParamFisL, ParamFisU, [], options, fisMg, [xlob, x2ob], yob,
    Msht);

fisMgOpt = F_changeFISmg(ParamFis_opt, fisMg, Msht); %
сист.нечёт.выв.после оптимиз.

% fuzzy(fisMgOpt)
% showfis(fisMgOpt)

%-----
% ПОСТРОЕНИЕ ГРАФИКА НЕЧЁТКОГО ОТОБРАЖЕНИЯ МАМДАНИ ПОСЛЕ АДАПТАЦИИ

yMgOpt = zeros(n, n);
for i = 1:n
    yMgOpt(i,:) = evalfis([x1; ones(size(x1))*x2(i)], fisMgOpt)';
end

%-----

h4 = figure(4);
set(h4,'Position',[1154 52 524 407])
colormap('default')
surf(x1, x2, yMgOpt)
axis([min(x1) max(x1) ...
    min(x2) max(x2) ...
    min(min(y)) max(max(y)) ]);
view(-40,30)
xlabel('x_1'); ylabel('x_2'); zlabel('y');
title('Система нечёткого вывода Мамдани после адаптации')

pause(2.5); % пауза 2,5 с

colormap('white') %
hold on
plot3(xlob, x2ob, yob, 'bs', 'MarkerSize',3,
'MarkerFaceColor','b')
plot3(xlts, x2ts, yts, 'ro', 'MarkerSize',3,
'MarkerFaceColor','r')
hold off

```

```

%-----
%-----
% Вычисление среднеквадратических значений ошибки нечёткой
аппроксимации
%-----

% FIS Мамдани до оптимизации
ytsMg = evalfis([x1ts, x2ts], fisMg); % реакция
FIS на тест.выборку
RMSE_Mg = sqrt(sum((yts-ytsMg).^2)/numel(yts)); %
среднекв.знач.ошибки

% FIS Мамдани после оптимизации
ytsMgOpt = evalfis([x1ts, x2ts], fisMgOpt); % реакция
FIS на тест.выборку
RMSE_MgOpt = sqrt(sum((yts-ytsMgOpt).^2)/numel(yts)); %
среднекв.знач.ошибки

disp(' ')
disp('Средние квадратические значения ошибки аппроксимации:')
disp([' - до адаптации: RMSE = ', num2str(RMSE_Mg)])
disp([' - после адаптации: RMSE = ', num2str(RMSE_MgOpt)])
disp(' ')

figure(2);
xlabel(['x_1 RMSE = ', num2str(RMSE_Mg)]);
figure(4);
xlabel(['x_1 RMSE = ', num2str(RMSE_MgOpt)]);

%-----
% close([h1 h2 h3 h4])

```

## 7. m-программы функций обновления параметров системы нечёткого вывода и определения ошибки аппроксимации:

```

function FISnew = F_changeFISmg(param, FISold, Msht)
% Установка новых параметров (param) нечеткой системы FISold

FISnew = FISold;

% Демасштабирование настраиваемых параметров
param = param ./ Msht;

% Весовые коэффициенты правил
FISnew.rule(1).weight = param(1);
FISnew.rule(2).weight = param(2);
FISnew.rule(3).weight = param(3);
FISnew.rule(4).weight = param(4);

```

```

FISnew.rule(5).weight = param(5);
FISnew.rule(6).weight = param(6);
FISnew.rule(7).weight = param(7);
FISnew.rule(8).weight = param(8);

% Коэффициенты концентрации термов входных переменных
FISnew.input(1).mf(1).params(1) = param(9);
FISnew.input(1).mf(2).params(1) = param(10);
FISnew.input(1).mf(3).params(1) = param(11);
FISnew.input(2).mf(1).params(1) = param(12);
FISnew.input(2).mf(2).params(1) = param(13);
FISnew.input(2).mf(3).params(1) = param(14);

% Коэффициенты концентрации термов выходных переменных
FISnew.output(1).mf(1).params(1) = param(15);
FISnew.output(1).mf(2).params(1) = param(16);
FISnew.output(1).mf(3).params(1) = param(17);
FISnew.output(1).mf(4).params(1) = param(18);
FISnew.output(1).mf(5).params(1) = param(19);

% коорд.максимумв некотор.термов входных переменных
FISnew.input(1).mf(2).params(2) = param(20);
FISnew.input(2).mf(2).params(2) = param(21);

% коорд.максимумв некотор.термов выходных переменных
FISnew.output(1).mf(2).params(2) = param(22);
FISnew.output(1).mf(3).params(2) = param(23);
FISnew.output(1).mf(4).params(2) = param(24);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function errFISmg = F_errFISmg(param, FISiter, inputFIS, target,
Msht)
% Расчет ошибки при новых параметрах (param) нечеткой системы
FISiter

% Установка новых параметров нечеткой системы
FISiter = F_changeFISmg(param, FISiter, Msht);

% Нечеткий вывод:
outFIS = evalfis(inputFIS, FISiter);

% Расчет ошибки:
errFISmg = sqrt(sum((target-outFIS).^2)/numel(outFIS)); %
среднекв.знач.ошибки

```



8-9.

График исходной зависимости согласно варианту задания:

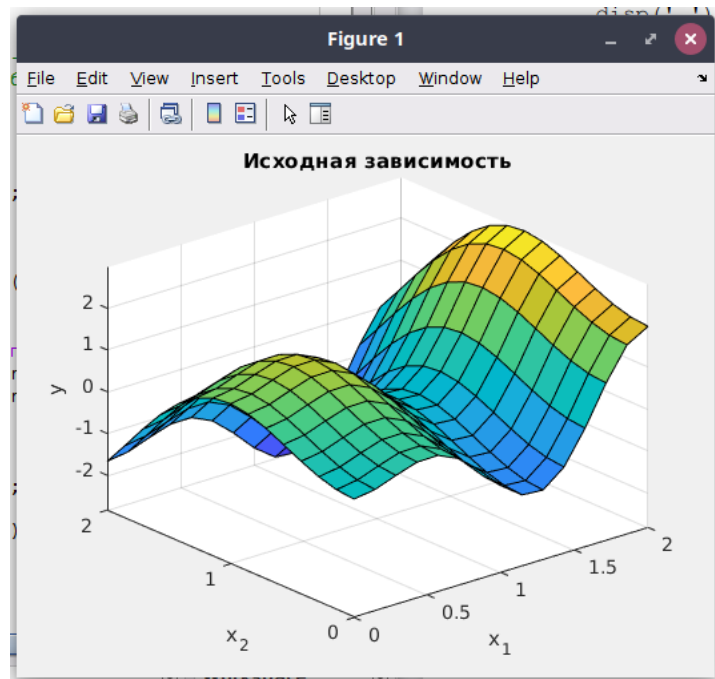


График распределения точек обучающей и тестовой выборки данных в области определения аппроксимируемой зависимости:

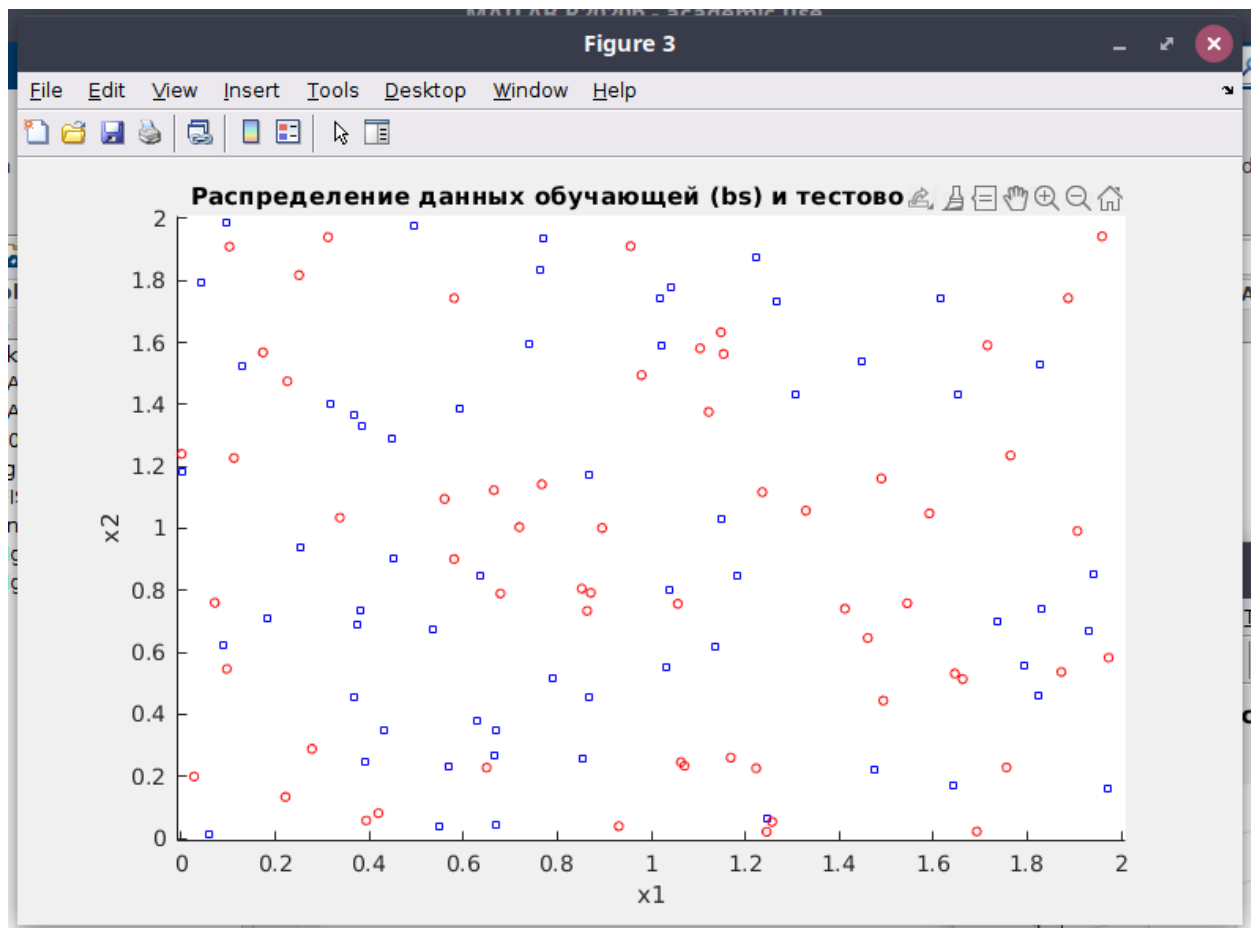


График: результат аппроксимации системой нечёткого вывода заданной зависимости до адаптации и среднее квадратическое значение ошибки аппроксимации до адаптации:

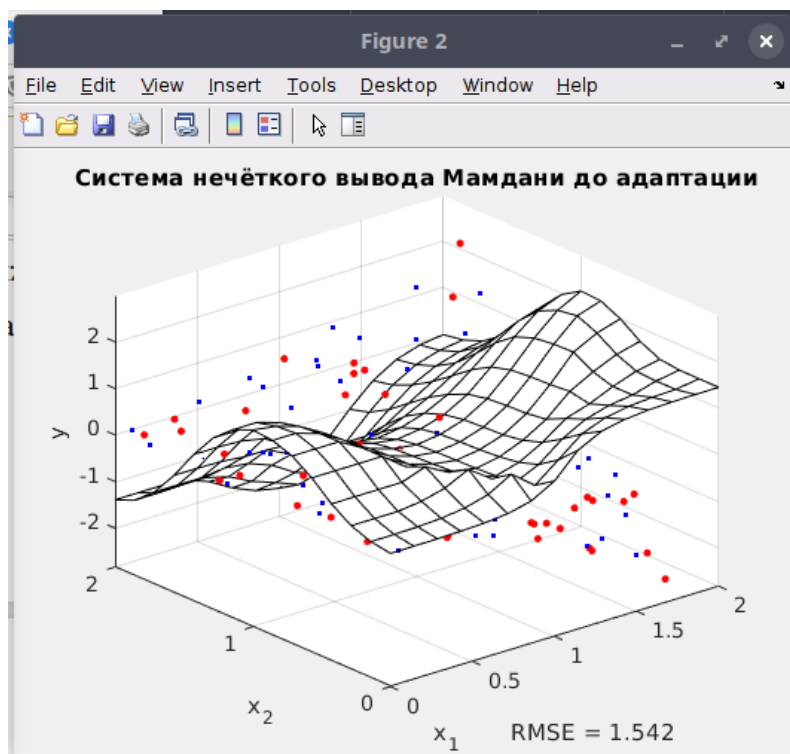
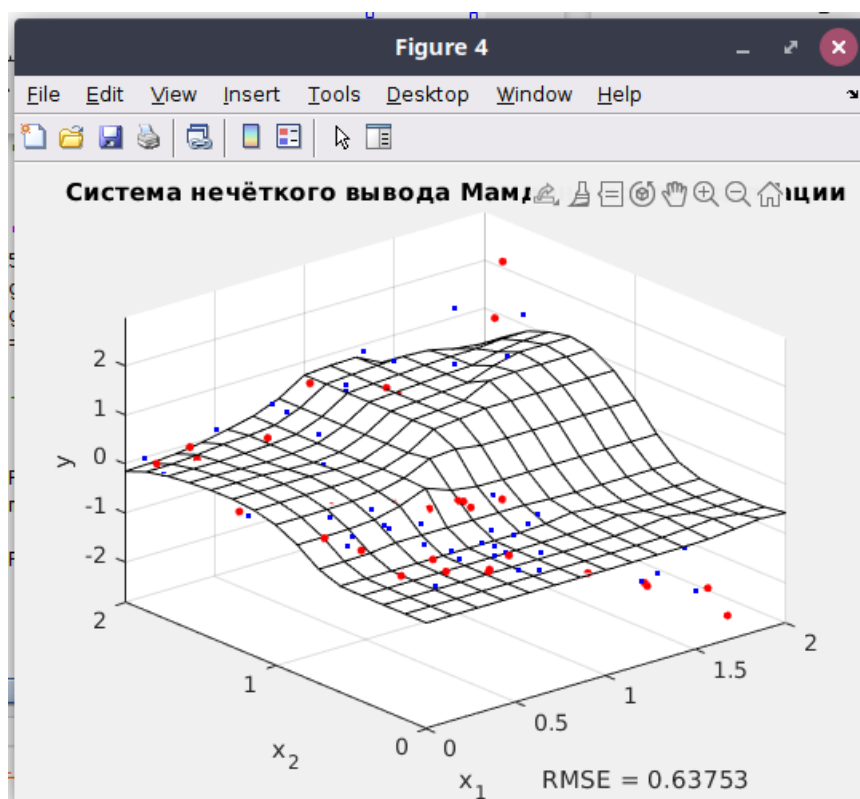
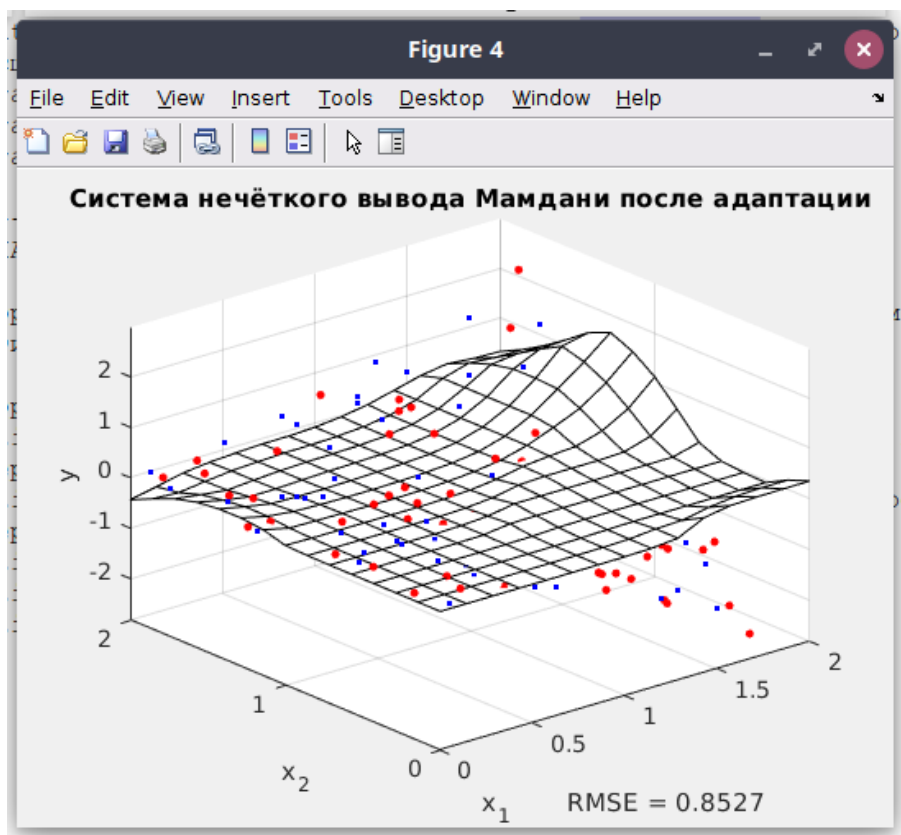


График: результат аппроксимации системой нечёткого вывода заданной зависимости после адаптации:

с использованием масштабирования настраиваемых параметров и среднее квадратическое значение ошибки аппроксимации после адаптации с использованием масштабирования настраиваемых параметров:



без использования масштабирования настраиваемых параметров и среднее квадратическое значение ошибки аппроксимации после адаптации без использования масштабирования настраиваемых параметров:



#### 10. Выводы по работе:

В ходе данной работы удалось получить ошибку аппроксимации 0,63753. Что меньше, чем ошибка исходной аппроксимации: 1,542; и ошибки лучшего результата работы 1: 0,69101.

Как и предполагалось – с использованием масштабирующих коэффициентов ошибка аппроксимации была меньше, чем без их использования: 0,8527.

Таким образом, можно утверждать, что с помощью пакета MATLAB Optimization Toolbox можно реально уменьшить ошибку аппроксимации.