Министерство образования и науки Российской Федерации Федеральное агентство по образованию

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Вятский государственный университет»

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

Отчет по лабораторной работе №8 дисциплины

«Системы обработки знаний»

Выполнил студент группы ИВТ-41 /Крючков И. С./ Проверил /Ростовцев В. С./

Киров 2024

1. Задание

Создать систему нечеткого вывода типа Мамдами, которая моделирует зависимость .

Диапазон x1: [-6; 5]

Диапазон x2: [-6; 4]

Построить график функции.

Выполнить моделирование со всеми типами функций принадлежности и выбрать ее оптимальный тип по критерию минимума среднеквадратичного отклонения.

Для алгоритма Мамдани выполнить моделирование со следующими параметрами:

* метод агрегации (максимум, сумма, вероятностное «ИЛИ»);
* метод дефаззификации (центр тяжести, медиана, наибольший из максимумов);
* метод «И» (минимум, умножение (вероятностное «И»));
* метод «ИЛИ» (максимум, вероятностное «ИЛИ»);
* метод импликации (минимум, умножение);

По результатам проектирования системы нечёткого вывода в Fuzzy Logic Toolbox составить отчет в электронном виде, включив в него результаты промежуточных этапов, а также графики моделируемой и аппроксимированной функции, сформировать выводы.

1. Ход работы

Графики функции, полученные в ходе лабораторной работы №7 представлены на рисунках 1-3.

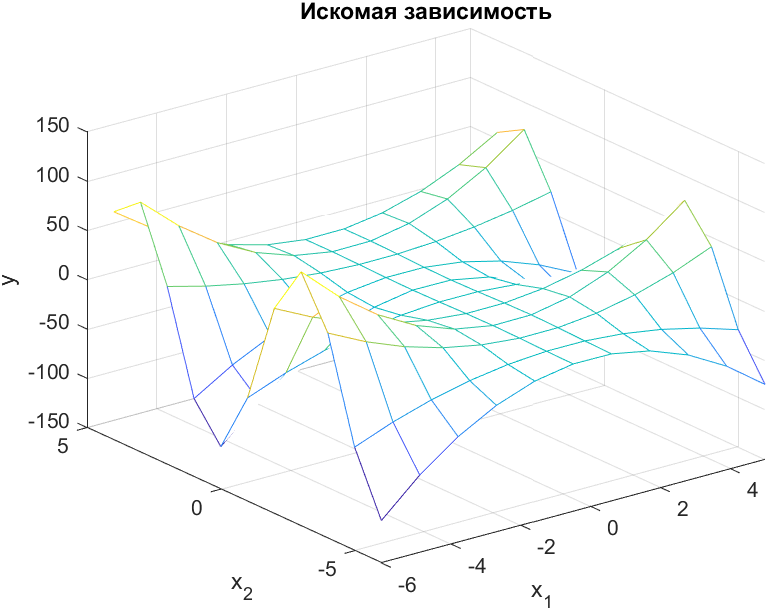


Рисунок 1 – График функции с 10 точками дискретизации

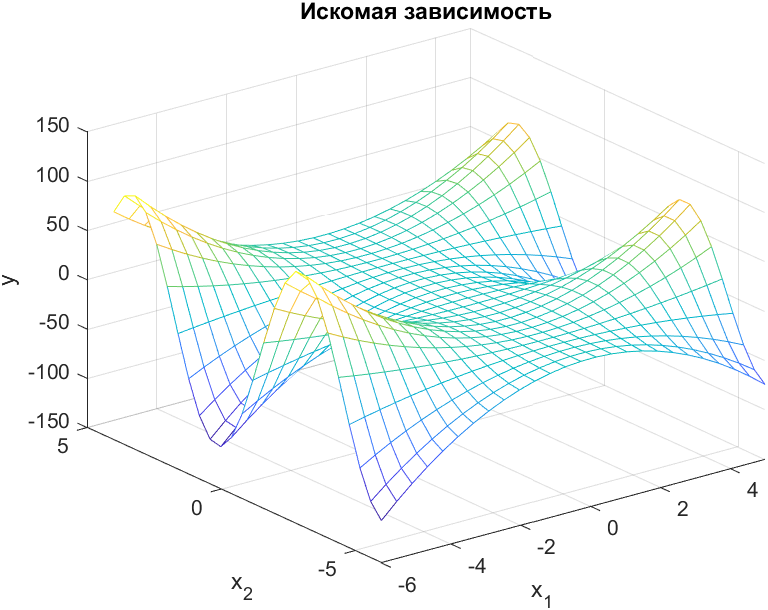


Рисунок 2 – График функции с 25 точками дискретизации

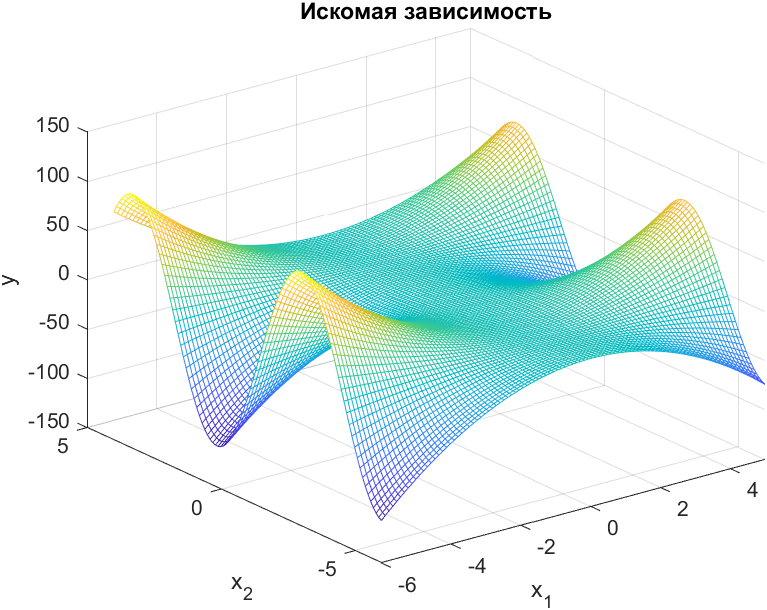


Рисунок 3 – График функции с 100 точками дискретизации

Нечеткие правила, полученные в ходе лабораторной работы №7:

Если X2 = «низкий» И X1 = «низкий», ТО y = «низкий»;

Если X2 = «низкий» И X1 = «средний», ТО y = «средний»;

Если X2 = «низкий» И X1 = «высокий», ТО y = «ниже среднего»;

Если X2 = «средний» И X1 = «низкий», ТО y = «средний»;

Если X2 = «средний» И X1 = «средний», ТО y = «средний»;

Если X2 = «средний» И X1 = «высокий», ТО y = «средний»;

Если X2 = «высокий» И X1 = «низкий», ТО y = «высокий»;

Если X2 = «высокий» И X1 = «средний», ТО y = «средний»;

Если X2 = «высокий» И X1 = «высокий», ТО y = «выше среднего»;

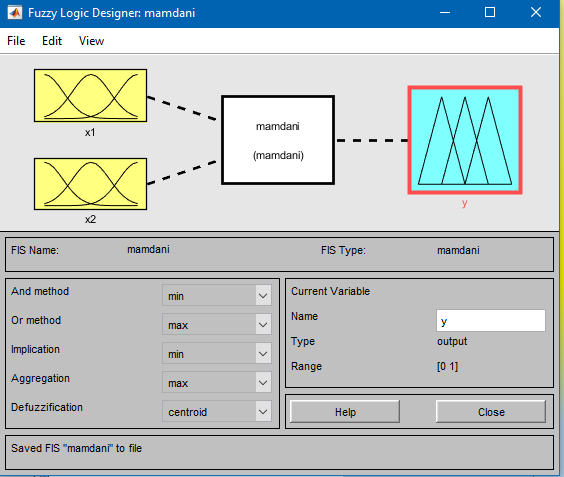


Рисунок 4 – Система Мамдами

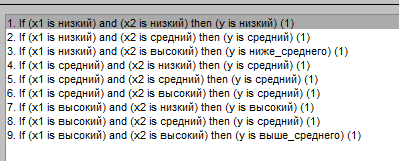


Рисунок 5 – Правила

Исходный код вычисления среднеквадратичной ошибки представлен на рисунке 6.

|  |
| --- |
| f = @(x1, x2)3\*x1.^2.\*cos(x2+3);  types = ["gaussmf", "gauss2mf", "trimf", "trapmf", "gbellmf"];  types\_errors = {};  for i = 1:length(types)  for j = 1:length(mamdani.inputs)  for k = 1:length(mamdani.inputs(j).mf)  mamdani.inputs(j).mf(k).type = types(i);  end  end  for j = 1:length(mamdani.outputs)  for k = 1:length(mamdani.outputs(j).mf)  mamdani.outputs(j).mf(k).type = types(i);  end  end    % fuzzyLogicDesigner(mamdani);  % pause;    [x1, x2, z] = gensurf(mamdani);    y = f(x1, x2);  E = immse(z, y);  disp(types(i) + " " + E);    types\_errors{end+1} = [types(i), E];  end  best\_type = types\_errors{1};  for i = 2:length(types\_errors)  if types\_errors{i}(2) < best\_type(2)  best\_type = types\_errors{i};  end  end  disp("Best type: " + best\_type(1) + " " + best\_type(2));  for j = 1:length(mamdani.inputs)  for k = 1:length(mamdani.inputs(j).mf)  mamdani.inputs(j).mf(k).type = best\_type(1);  end  end  for j = 1:length(mamdani.outputs)  for k = 1:length(mamdani.outputs(j).mf)  mamdani.outputs(j).mf(k).type = best\_type(1);  end  end  fuzzyLogicDesigner(mamdani); |

Рисунок 6 – Код вычисления среднеквадратичной ошибки

Таблица 1 – результаты ошибок при изменении типа входов и выхода

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| gaussmf | gauss2mf | trimf | trapmf | gbellmf |
| 2755.15 | 3373.2322 | 2954.3917 | 3216.8576 | 3434.2117 |

Наибольшей точностью обладает функция gaussmf. График функции с применением gaussmf для входов и выходов представлен на рисунке 7.

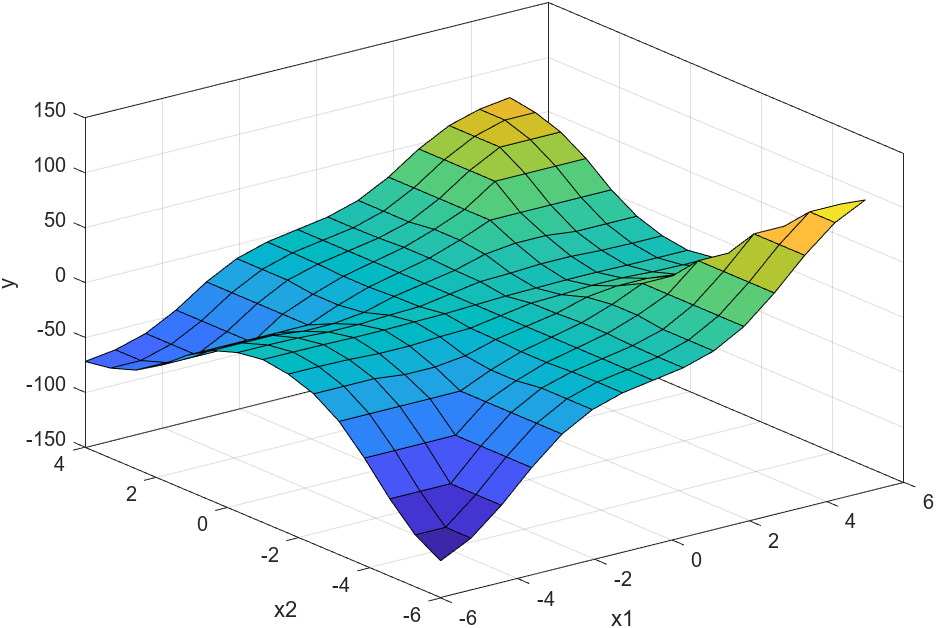


Рисунок 7 – График функции с применением gaussmf для входов и выходов

|  |
| --- |
| f = @(x1, x2)3\*x1.^2.\*cos(x2+3);  best\_type = "trimf";  for i = 1:length(mamdani.inputs)  for j = 1:length(mamdani.inputs(i).mf)  mamdani.inputs(i).mf(j).type = best\_type;  end  end  for i = 1:length(mamdani.outputs)  for j = 1:length(mamdani.outputs(i).mf)  mamdani.outputs(i).mf(j).type = best\_type;  end  end  % fuzzyLogicDesigner(mamdani);  % pause;  and\_methods = ["min", "prod"];  or\_methods = ["max", "probor"];  implication\_methods = ["min", "prod"];  aggregation\_methods = ["max", "probor"];  defuzz\_methods = ["centroid", "bisector", "lom", "mom", "som"];  best\_and\_method = "";  best\_and\_error = intmax;  for i = 1:length(and\_methods)  mamdani.AndMethod = and\_methods(i);  [x1, x2, z] = gensurf(mamdani);    y = f(x1, x2);  E = immse(z, y);  disp(and\_methods(i) + " error: " + E);  if E < best\_and\_error  best\_and\_error = E;  best\_and\_method = and\_methods(i);  end  end  mamdani.AndMethod = best\_and\_method;  best\_or\_method = "";  best\_or\_error = intmax;  for i = 1:length(or\_methods)  mamdani.OrMethod = or\_methods(i);    [x1, x2, z] = gensurf(mamdani);    y = f(x1, x2);  E = immse(z, y);  disp(or\_methods(i) + " error: " + E);  if E < best\_or\_error  best\_or\_error = E;  best\_or\_method = or\_methods(i);  end  end  mamdani.OrMethod = best\_or\_method;  best\_implication\_method = "";  best\_implication\_error = intmax;  for i = 1:length(implication\_methods)  mamdani.ImplicationMethod = implication\_methods(i);    [x1, x2, z] = gensurf(mamdani);    y = f(x1, x2);  E = immse(z, y);  disp(implication\_methods(i) + " error: " + E);  if E < best\_implication\_error  best\_implication\_error = E;  best\_implication\_method = implication\_methods(i);  end  end  mamdani.ImplicationMethod = best\_implication\_method;  best\_aggregation\_method = "";  best\_aggregation\_error = intmax;  for i = 1:length(aggregation\_methods)  mamdani.AggregationMethod = aggregation\_methods(i);    [x1, x2, z] = gensurf(mamdani);    y = f(x1, x2);  E = immse(z, y);  disp(aggregation\_methods(i) + " error: " + E);  if E < best\_aggregation\_error  best\_aggregation\_error = E;  best\_aggregation\_method = aggregation\_methods(i);  end  end  mamdani.AggregationMethod = best\_aggregation\_method;  best\_defuzz\_method = "";  best\_defuzz\_error = intmax;  for i = 1:length(defuzz\_methods)  mamdani.DefuzzMethod = defuzz\_methods(i);    [x1, x2, z] = gensurf(mamdani);    y = f(x1, x2);  E = immse(z, y);  disp(defuzz\_methods(i) + " error: " + E);  if E < best\_defuzz\_error  best\_defuzz\_error = E;  best\_defuzz\_method = defuzz\_methods(i);  end  end  mamdani.DefuzzMethod = best\_defuzz\_method;  fuzzyLogicDesigner(mamdani);  pause;  [x1, x2, z] = gensurf(mamdani);  y = f(x1, x2);  E = immse(z, y);  disp("Final error: " + E); |

Рисунок 8 – Исходный код поэтапного подбора параметров

Таблица 2 – Результат поэтапного подбора

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| and | | or | | implication | | aggregation | | defuzzification | | | | |
| min | prod | max | probor | min | probor | max | probor | centroid | bisector | lom | mom | som |
| 2761.14 | 2769.57 | 2761.14 | 2761.14 | 2761.14 | 2881.24 | 2954.50 | 2761.14 | 2761.14 | 2906.49 | 5248.57 | 3979.68 | 3476.80 |

Лучшие параметры представлены на рисунке 9.

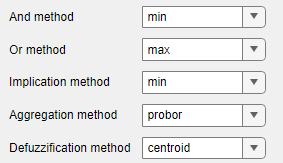


Рисунок 9 – Лучшие параметры методом поэтапного подбора

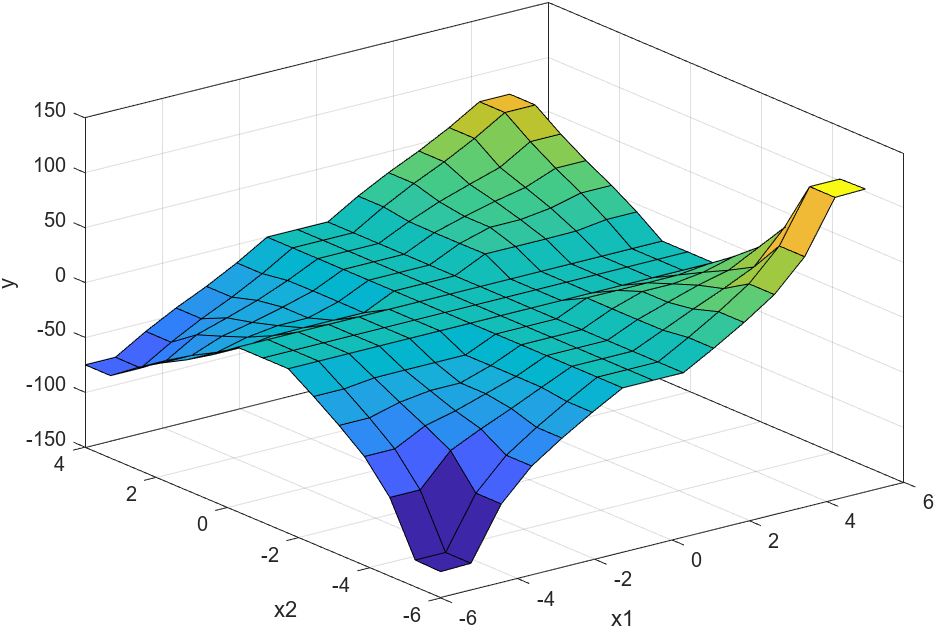


Рисунок 10 – График функции, полученный в результате поэтапного подбора параметров

Итоговая среднеквадратичная ошибка составила: 2761.1466.

Выводы

В ходе лабораторной работы была составлена система Мамдани.

Были подобраны функции, а также параметры моделирования.

Был написан скрипт для поэтапного подбора параметров.

Результат экспериментов показал, что наибольшую эффективность среди используемых функций принадлежности имеет функция gaussmf. Среднеквадратичная ошибка этой функции составила 2755.15, наихудший результат показала функция gbellmf со среднеквадратичной ошибкой 3434.2117.

Среднеквадратичная ошибка имеет большое значение предположительно из-за малого количества правил.

Полный перебор, реализованный скриптом, позволил подобрать лучшие параметры аппроксимации (рис. 13).

После подбора оптимальных параметров среднеквадратичная ошибка составила 231.5297.