Краткое введение в нечеткий вывод.

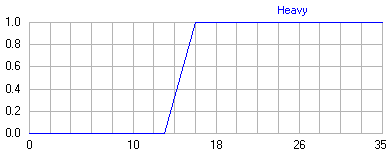
Определения:

Нечеткая переменная – это кортеж вида <α, X, Α>, где:

α – имя нечеткой переменной; X – её область определения;

A – нечеткое множество на универсуме X.

Пример: Нечеткая переменная <«Тяжелый бронежилет», {x| 0 кг < x < 35 кг}, B={x, μ(x)}> характеризует массу военного бронежилета. Будем считать его тяжелым, если его масса > 16 кг, риссунок 1.3.

  
Рисунок 1.3 – График функции принадлежности μ(x) для нечеткого множества B.

Лингвистическая переменная есть кортеж <β, T, X, G, M>, где:

β – имя лингвистической переменной;

T – множество её значений (термов);

X – универсум нечетких переменных;

G – синтаксическая процедура образования новых термов;

M – семантическая процедура, формирующая нечеткие множества для каждого терма данной лингвистической переменной.

Пример: Допустим, мы имеем субъективную оценку массы бронежилета. Она, например, может быть получена от военнослужащих (выступающих в роли экспертов), которые непосредственно имеют дело с подобной амуницией. Формализовать эту оценку можно с помощью следующей лингвистической переменной <β, T, X, G, M> находящийся на рисунке 1.4, где:

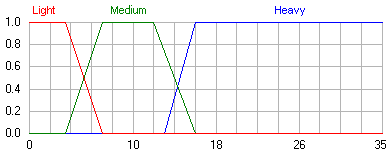
β – Бронежилет;

T – {«Легкий бронежилет (Light)», «Бронежилет средней массы (Medium)», «Тяжелый бронежилет (Heavy)»};

X = [0; 35];

G – процедура образования новых термов при помощи логических связок и модификаторов. Например, «очень тяжелый бронежилет»;

M – процедура задания на универсуме X=[0; 35] значений лингвистической переменной, т.е. термов из множества T.

  
Рисунок 1.4 – Графики функций принадлежности значений лингвистической переменной «Бронежилет».

Нечетким высказыванием будем называть высказывание вида "β IS α", где:

β *–* лингвистическая переменная;

α *–* один из термов этой переменной.

Пример: «Бронежилет IS легкий». Здесь «Бронежилет» – это лингвистическая переменная, а «легкий» её значение.

Упрощенно говоря, правилом нечетких продукций – это классическое правило вида «ЕСЛИ… ТО ...», где в качестве условий и заключений используются нечеткие высказывания. Записываются такие правила в следующем виде:

IF (β1 IS α1) AND (β2 IS α2) THEN (β3 IS α3).

Кроме «AND» также используются логическая связка «OR». Также каждое из нечетких высказываний в условии любого правила будем называться подусловием. Аналогично, каждое из высказываний в заключении называется подзаключением.

Напомним, как выглядит высказываний для модели Мамдани:

1) IF (Бронежилет тяжелый) THEN (Солдат уставший);

2) IF (Муж трезвый) AND (Зарплата высокая) THEN (Жена довольная).

Высказывание для модели Сугено анологично Мамдани, за исключением заключений правил, которые задаются не термами, а линейной функцией вида: , где j – номер правила, i – порядок модели Сугено. В самом простом случае модель может быть 0 порядка, тоесть заключение будет константа .

**Описание задачи из примера tipper в Matlab**

Находясь в ресторане города Санкт-Петербург штата Флорида согласно принятым в США традициям, после окончания обслуживания посетителя принято оставлять официанту чаевые. На основе местных обычаев и интуиции посетителей, величина чаевых не является постоянной и зависит, например, от качества обслуживания (service) и качества принятых блюд (food).

Задача в том, чтобы разработать экспертную систему нечеткого вывода позволяющую определять величину чаевых.

Стартовой точкой для построения экспертной системы являются три золотых правила основанные на многолетнем опыте посещения ресторанов.

* 1. Если обслуживание плохое (poor) или ужин стух/подгорел (rancid), то чаевые малые (cheap).
  2. Если обслуживание хорошее (good), то чаевые средние (average).
  3. Если обслуживание отличное (excellent) или ужин превосходный (delicious), то чаевые щедрые (generous).

В качестве входных параметров системы нечеткого вывода в примере используются две нечеткие лингвистические переменные: service и food. а в качестве выходной переменную tip (величина чаевых). В качестве терм-множеств:

1. T1(service) = {poor, good, excellent};
2. T2(food) = {rancid, delicious};
3. T3(tip) = {cheap, average, generous}.

При этом каждый из термов входных переменных будем оценивать по 10-балльной шкале, а термы выходной переменной – в процентах от стоимости заказанных блюд. Будем считать, что для переменной tip значение average равно 15%, generous равно 25%, а значение терма cheap равно 5%. В результате получаем график результата обработки терм-множества расположеный на риссунке 1.5.



Рисунок 1.5 – График результата обработки терм-множества tip.

Очевидно, что число и форма кривой зависят от местных традиции, достатка клиента, и так далее, но правила довольно универсальны. Из всего этого следует, что в нем используются система нечеткого вывода Мамдани.

Лабораторная работа №5

Программная реализация Упрощенного

частного случая вывода мамдани

Цель работы:

Повторение ранее изученного вывода Мамдани и программной реализация этапа вывода.

Требования к отчёту

Отчёт должен содержать:

1. Листинги программы, созданной при выполнении лабораторной работы.
2. Термы и правила, составленные в Matlab для своего варианта.
3. Таблицу, содержащую 50 выходных значений, полученных из программной реализации вывода и используя Matlab.

задание на выполнение лабараторной работЫ

1. Программная реализация частного случая вывода Мамдани.

* 1. Прочитать справочную информацию об алгоритме Мамдани.
  2. Реализовать вывод Мамдани в соответствии с вариантами задания.
  3. Реализовать FIS – типа Мамдани в соответствии с вариантом используя Fuzzy Tool Box.
  4. Сравнить выходные значения программной реализации вывод Мамдани и полученные в Matlab в 50 точках, распределенных на всей области терм-множества с постоянным шагом.

Примечание: выбранный язык для выполнения лабораторной работы обязательно должен быть ООП. В работе необходимо минимум реализовать следующие классы:

1. Абстрактный класс Term и наследуемые от него термы необходимые для выполнения работы
2. Абстрактный класс Norm и наследуемые от него абстрактные классы TNorm, SNorm, а также наследуемые от них классы норм.
3. Абстрактный класс Defuzzifier и класс используемого метода дефаззификации.
4. Результат выдаваемый программой должен быть округлен до 2 знаков после точки.

задания по ВАРИАНТАМ

Таблица 1 – правила нечеткого вывода.

|  |  |
| --- | --- |
| Вар. | Правила |
| 1 | Если T11 и T22 и T31 То X1;  Если T12 или T23 или T32 То X2;  Если T13 и T21 То X3; |
| 2 | Если T11 и T23 и T32 То X1;  Если T32 или T22 или T13 То X2;  Если T12 и T31 То X3; |
| 3 | Если T21 и T13 и T32 То X1;  Если T12 или T23 или T31 То X2;  Если T11 и T22 То X3; |
| 4 | Если T11 и T21 и T32 То X2;  Если T22 и T13 или T31 То X1;  Если T12 или T23 То X3; |
| 5 | Если T11 и T23 и T31 То X2;  Если T11 или T22 или T32 То X1;  Если T13 и T21 То X3; |
| 6 | Если T21 и T13 и T32 То X1;  Если T12 и T22 и T33 То X2;  Если T11 или T31 То X3; |
| 7 | Если T31 и T11 и T21 То X1;  Если T12 или T31 или T23 То X3;  Если T13 и T22 То X3; |
| 8 | Если T12 и T23 и T31 То X3;  Если T21 и T32 и T11 То X2;  Если T22 и T13 То X1; |
| 9 | Если T31 или T22 или T12 То X1;  Если T32 и T13 и T21 То X3;  Если T11 или T23 То X2; |
| 10 | Если T12 и T31 и T21 То X1;  Если T22 или T32 или T11 То X2;  Если T13 и T23 То X3; |
| 11 | Если T31 и T33 и T23 То X2;  Если T11 и T32 и T22 То X3;  Если T21 и T12 То X1; |
| 12 | Если T13 и T21 и T32 То X1;  Если T22 или T31 или T11 То X2;  Если T12 и T23 То X3; |
| 13 | Если T12 и T31 и T22 То X3;  Если T21 или T13 или T11 То X2;  Если T23 и T13 То X1; |
| 14 | Если T11 и T32 и T22 То X3;  Если T12 и T31 и T23 То X1;  Если T13 и T21 То X2; |
| 15 | Если T12 и T21 и T32 То X1;  Если T13 и T22 и T31 То X3;  Если T23 и T11 То X2; |
| 16 | Если T11 или T22 То X2;  Если T12 и T21 То X1;  Если T23 и T14 То X3; |
| 17 | Если T12 и T23 То X1;  Если T21 и T11 и T32 То X3;  Если T22 и T11 и T31 То X2; |
| 18 | Если T12 или T22 или T31 То X3;  Если T23 и T13 и T32 То X2;  Если T21 и T11 То X1; |
| 19 | Если T11 и T21 и T32 То X1;  Если T12 или T22 или T31 То X2;  Если T13 и T23 То X3; |
| 20 | Если T12 или T31 или T23 То X1;  Если T21 и T32 и T11 То X3;  Если T33 и T22 и T13 То X2; |
| 21 | Если T13 и T32 и T21 То X1;  Если T12 или T22 или T31 То X3;  Если T11 и T23 То X2; |
| 22 | Если T12 и T31 и T22 То X1;  Если T23 или T23 или T13 То X3;  Если T31 и T21 То X2; |
| 23 | Если T21 и T13 и T32 То X1;  Если T12 и T23 и T31 То X2;  Если T11 и T22 То X3; |
| 24 | Если T11 и T21 и T32 То X3;  Если T22 и T13 и T31 То X1;  Если T21 и T23 То X2; |
| 25 | Если T33 и T21 и T12 То X1;  Если T22 или T31 или T11 То X2;  Если T23 и T32 То X3; |

Таблица 2 – указания к значениям T1, T2 и т.д из таблицы 1.

|  |  |
| --- | --- |
| Вар. | Функции принадлежности термов |
| 1 | T1x – B, T2x – B, T3x – SD, X1 – B, X2 – Trap, X3 – Trian. |
| 2 | T1x – B , T2x – G, T3x – B, X1 – G, X2 – G, X3 – G. |
| 3 | T1x – PI-S, T2x – Trian, T3x – Trian, X1 – Trian, X2 – Trian,  X3 – Trian. |
| 4 | T1x – B , T2x – G, T3x – B, X1 – G, X2 – G, X3 – G. |
| 5 | T1x – B , T2x – B, T3x – G, X1 – G, X2 – G, X3 – B. |
| 6 | T1x – B, T2x – B, T3x – B, X1 – PI-S, X2 – PI-S, X3 – PI-S. |
| 7 | T1x – G , T2x – G, T3x – Ttrap, X1 – G, X2 – G, X3 – Ttrap. |
| 8 | T1x – SD, T2x – SD, T3x – SD, X1 – G, X2 – G, X3 – Ttrap. |
| 9 | T1x – Ttrap , T2x – Trian, T3x – Ttrap, X1 – G, X2 – G, X3 – Ttrap. |
| 10 | T1x – SD , T2x – PI-S, T3x – PI-S, X1 – G, X2 – G, X3 – Ttrap. |
| 11 | T1x – B , T2x – Ttrap, T3x – Ttrap, X1 – G, X2 – G, X3 – Ttrap. |
| 12 | T1x – G , T2x – B, T3x – B, X1 – G, X2 – G, X3 – G. |
| 13 | T1x – G , T2x – PI-S, T3x – Ttrap, X1 – G, X2 – G, X3 – Ttrap. |
| 14 | T1x – Trian, T2x – Trian, T3x – Trian, X1 – Trian, X2 – Trian, X3 – Trian. |
| 15 | T1x – PI-S, T2x – Ttrap, T3x – Ttrap, X1 – G, X2 – G, X3 – Trian. |
| 16 | T1x – Ttrap , T2x – Ttrap, T3x – Trian, X1 – G, X2 – G, X3 – Ttrap. |
| 17 | T1x – Ttrap , T2x – Ttrap, T3x – Ttrap, X1 – G, X2 – G, X3 – Ttrap. |
| 18 | T1x – B , T2x – B, T3x – Ttrap, X1 – G, X2 – G, X3 – Ttrap. |
| 19 | T1x – Ttrap , T2x – Ttrap, T3x – G, X1 – G, X2 – G, X3 – Ttrap. |
| 20 | T1x – B , T2x – Ttrap, T3x – Ttrap, X1 – G, X2 – G, X3 – Ttrap. |
| 21 | T1x – Ttrap , T2x – G, T3x – Ttrap, X1 – G, X2 – G, X3 – Ttrap. |
| 22 | T1x – Ttrap , T2x – Trian, T3x – PI-S, X1 – G, X2 – G, X3 – Trian. |
| 23 | T1x – Ttrap , T2x – Ttrap, T3x – Ttrap, X1 – G, X2 – G, X3 – Ttrap. |
| 24 | T1x – SD , T2x – SD, T3x – SD, X1 – SD, X2 – Ttrap, X3 – Ttrap. |
| 25 | T1x – Ttrap , T2x – B, T3x – B, X1 – Trian, X2 – Trian, X3 – Ttrap. |

T11, T21, …, T21 и т.д. являются термами, где первая цифра означает, номер переменной частью которой является терм, а вторая цифра номер терма. Термы имеют произвольные значения, но обязательно должна быть возможность выполнится всем правилам, так же X1>X2>X3, T13>T12>T11, T23>T22 и т.д.

В таблице 2 представлены сокращения функций принадлежности (для расшифровки см. приложение). Implication, And, Or Aggregation необходимо выбрать на своё усмотрение, defuzzification выбрать centroid.

Справочная информация.

Краткое описание вывода Мамдани

Алгоритм Мамдани.

Данный алгоритм описывает несколько последовательно выполняющихся этапов, расположенных на рисунке 3.1. При этом каждый последующий этап получает на вход значения полученные на предыдущем шаге.

d3963d1f04f1e048aca0db4ed759cdb2

Рисунок 3.1 – Диаграмма деятельности процесса нечеткого вывода

*1. Формирование базы правил*

База правил — это множество правил, где каждому подзаключению сопоставлен определенный весовой коэффициент.

База правил может иметь следующий вид (для примера используются правила различных конструкций):

RULE\_1: IF «Condition\_1» THEN «Conclusion\_1» (F1) AND «Conclusion\_2» (F2);

RULE\_2: IF «Condition\_2» AND «Condition\_3» THEN «Conclusion\_3» (F3);

…

RULE\_n: IF «Condition\_k» THEN «Conclusion\_(q-1)» (Fq-1) AND «Conclusion\_q» (Fq);

Где Fi — весовые коэффициенты, означающие степень уверенности в истинности получаемого подзаключения (i = 1..q). По умолчанию весовой коэффициент принимается равным 1. Лингвистические переменные, присутствующие в условиях называются входными, а в заключениях выходными.

Обозначения:

n — число правил нечетких продукций.

m — кол-во входных переменных.

s — кол-во выходных переменных.

k — общее число подусловий в базе правил.

q — общее число подзаключений в базе правил.

Примечание: Данные обозначения будут использоваться в последующих этапах.

*2. Фаззификация входных переменных (приведение к нечеткости)*

На вход поступают сформированная база правил и массив входных данных А = {a1, ..., am}. В этом массиве содержатся значения всех входных переменных. Целью этого этапа является получение значений истинности для всех подусловий из базы правил. Это происходит так: для каждого из подусловий находится значение bi = μ(ai). Таким образом получается множество значений bi (i=1..k).

*3. Агрегирование подусловий*

Как уже упоминалось выше, условие правила может быть составным, т.е. включать подусловия, связанные между собой при помощи логической операции. Целью этого этапа является определение степени истинности условий для каждого правила системы нечеткого вывода.

*Проще говоря* этап 2 и 3 заключается в раскрытии условий используя максиминный или колорометрический подход реализации дизъюнкции и конюнкции. Вспомним, что есть два основных способа реализации дизъюнкции и конъюнкции:

Максиминный подход: Колорометрический подход:

a || b => max(a, b) a || b => a + b - a \* b

a && b => min(a, b) a && b => a \* b

Отрицание задается как: !а => 1-a.

*4. Активизация подзаключений*

На этом этапе происходит переход от условий к подзаключениям. Существует несколько вариантов выполнения данного этапа, например, импликация prod - активизация (Ларсена) и min – активизация (Мамдани) которую рассмотрим далее. Для каждого подзаключения находится степень истинности di = ci\*Fi, где i = 1..q. Затем, опять же каждому i-му подзаключению, сопоставляется множество Di с новой функцией принадлежности. Её значение определяется как минимум из di и значения функции принадлежности терма из подзаключения. Этот метод называется min-активизацией, который формально записывается следующим образом:

μ'i(x) = min {di, μi(x)}.

Где:

μ'i(x) — «активизированная» функция принадлежности;

μi(x) — функция принадлежности терма;

di — степень истинности i-го подзаключения.

Тоесть цель этого этапа – это получение совокупности «активизированных» нечетких множеств Di для каждого из подзаключений в базе правил (i = 1..q).

Конечный результат выше описанных действий после агрегирования на примере структуры tipper расположен на риссунке 3.2 «справа».

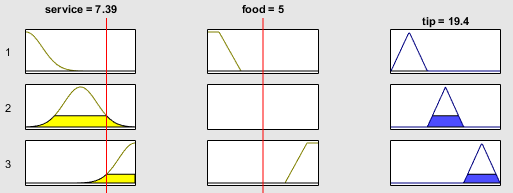


Рисунок 3.2 – Результат фазификации входных переменных и агрегирования подусловий

*5. Аккумулирование заключений*

Целью этого этапа является получение нечеткого множества для каждой из выходных переменных. Выполняется он следующим образом: i-ой выходной переменной сопоставляется объединение множеств Ei = ∪ Dj. Где j – номера подзаключений в которых участвует i-aя выходная переменная (i = 1..s). Объединением двух нечетких множеств является третье нечеткое множество со следующей функцией принадлежности:

μ'i(x) = max {μ1(x), μ2(x)}, где μ1(x), μ2(x) — функции принадлежности объединяемых множеств.

Результат аккумуляции подзаключений для структуры tipper находится на рисунке 3.3.

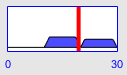
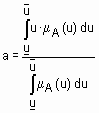


Рисунок 3.3 – Результат акумуляции подзаключений для структуры tipper.

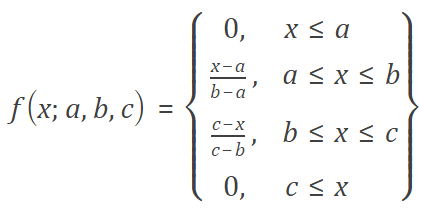
*6. Дефаззификация выходных переменных*

Цель дефаззификациии получить количественное значение (crisp value) для каждой из выходных лингвистических переменных. Существует несколько способов дефаззификации, рассмотрим метод центра тяжести.

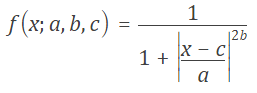
Дефаззификация нечеткого множества по методу центра тяжести осуществляется по формуле .

В случае дискретного универсального множества дефаззификация нечеткого множества  по методу центра тяжести осуществляется по формуле , по сути являющейся численным интегрированием.

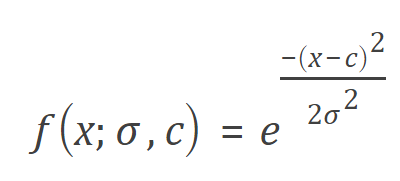
Контрольные вопросы

1. Типы FIS-структур, реализованные в Fuzzy Logic Toolbox
2. Этапы процедуры нечёткого логического вывода Мамдани
3. Понятие треугольной нормы (T-нормы)
4. Понятие треугольной конормы (S-нормы)
5. Нечёткая импликация, определение
6. Получение значений выходных переменных по заданным входам для вывода Мамдани
7. Как выглядит пример процедуры дефаззификации методом центра масс для треугольного терма в шести альфа уровнях (альфа уровни – 0.1, 0.3, 0.5, 0.7, 0.9, 1. Для нахождения значения альфа уровня треугольного терма использовать уравнение: 
8. Основные способы нахождения заключений при выводе Мамдани.
9. Способы определения параметров функций принадлежности.
10. Лингвистическая переменная, её составляющие
11. Преимущества и недостатки Сугэно-FIS по сравнению с Мамдани-FIS.

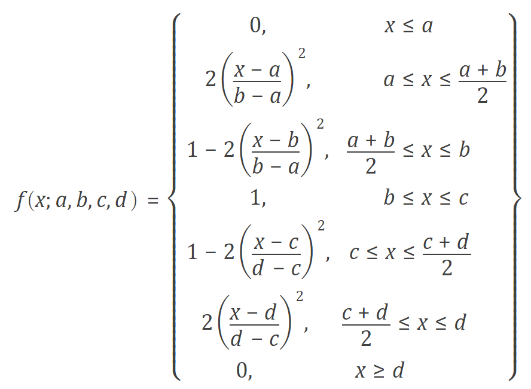
Приложение. Необходимы для лабараторной 5 функции.



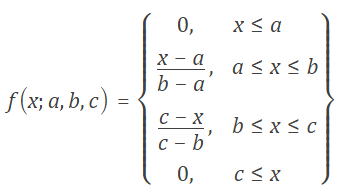
Bell(B) функция:



Gauss(G) функция:



Pi-shaped(PI-S) функция:



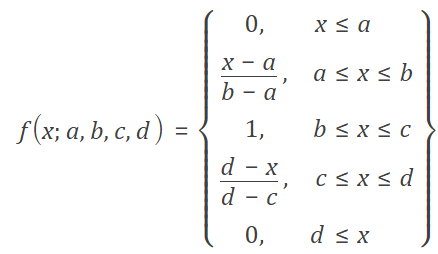
Triangular-shaped(Trian) функция:

Sigmidal Difference(SD) функция:

1) sigmoid функция:



2) difference функция:



Trapezoidal-shaped(Trap) функция: