

ПРИЛОЖЕНИЕ А
ФОНД ОЦЕНОЧНЫХ МАТЕРИАЛОВ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ
ПО ДИСЦИПЛИНЕ «Применение интеллектуальных технологий при проектировании
элементов систем защиты информации»

1. Перечень оценочных средств для компетенций, формируемых в результате освоения дисциплины

| Код контролируемой компетенции | Способ оценивания | Оценочное средство |
|--|-------------------|---|
| ПК-2: Способен проектировать элементы системы защиты объектов информатизации | Экзамен | Комплект контролирующих материалов для экзамена |

2. Описание показателей и критериев оценивания компетенций, описание шкал оценивания

Оцениваемые компетенции представлены в разделе «Перечень планируемых результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций» рабочей программы дисциплины «Применение интеллектуальных технологий при проектировании элементов систем защиты информации».

При оценивании сформированности компетенций по дисциплине «Применение интеллектуальных технологий при проектировании элементов систем защиты информации» используется 100-балльная шкала.

| Критерий | Оценка по 100-балльной шкале | Оценка по традиционной шкале |
|---|------------------------------|------------------------------|
| Студент освоил изучаемый материал (основной и дополнительный), системно и грамотно излагает его, осуществляет полное и правильное выполнение заданий в соответствии с индикаторами достижения компетенций, способен ответить на дополнительные вопросы. | 75-100 | <i>Отлично</i> |
| Студент освоил изучаемый материал, осуществляет выполнение заданий в соответствии с индикаторами достижения компетенций с не принципиальными ошибками. | 50-74 | <i>Хорошо</i> |
| Студент демонстрирует освоение только основного материала, при выполнении заданий в соответствии с индикаторами достижения компетенций допускает отдельные ошибки, не способен систематизировать материал и делать выводы. | 25-49 | <i>Удовлетворительно</i> |
| Студент не освоил основное содержание изучаемого материала, задания в соответствии с | <25 | <i>Неудовлетворительно</i> |

| | | |
|---|--|--|
| индикаторами достижения компетенций не выполнены или выполнены неверно. | | |
|---|--|--|

3. Типовые контрольные задания или иные материалы, необходимые для оценки уровня достижения компетенций в соответствии с индикаторами

1.Задания на применение технологий искусственного интеллекта при проектировании и разработке программных элементов системы защиты объектов информатизации

| Компетенция | Индикатор достижения компетенции |
|---|--|
| ПК-2 Способен проектировать элементы системы защиты объектов информатизации | ПК-2.1 Применяет инструментальные средства и технологии при проектировании и разработке элементов системы защиты объектов информатизации |

Тест 1

Имеется два игрока, каждый из которых стремится достичь выигрыша, представляющего собой сумму капиталовложений фирмы в новую систему защиты объекта информатизации на предприятии. Найти аналитическое решение игры 2x2 решением системы 3 уравнений, предварительно проверив наличие или отсутствие седловой точки. Результаты расчета будут использоваться для проектирования системы защиты. Требуется применить инструментальные средства языка Python для разработки программы нахождения решения поставленной задачи.

Пример решения

Матрица выигрышей

7 6
4 10

Проверено существование седловой точки: она отсутствует, т.к. нижняя оценка (6) не равна верхней оценке (7).

Составлена система трех уравнений, для нахождения p_1 и p_2 :

Запишем систему уравнений.

Для игрока I

$$7p_1 + 4p_2 = S$$

$$6p_1 + 10p_2 = S$$

$$p_1 + p_2 = 1$$

Для игрока II

$$7q_1 + 6q_2 = S$$

$$4q_1 + 10q_2 = S$$

$$q_1 + q_2 = 1$$

Решая эти системы находим:

$$S = 6,57$$

$$p_1 = 0,86$$

$$p_2 = 0,14$$

$$q_1 = 0,57$$

$$q_2 = 0,43$$

Задача

матрица выигрышей

4 0
3 9

Тест 2

Для заданной матрицы игры, связанной со стратегиями проектирования системы защиты объекта информатизации, свести задачу к задаче линейного программирования и построить симплекс таблицу после одного пересчета. Результаты расчета будут использоваться для проектирования системы защиты. Требуется применить инструментальные средства языка Python для разработки программы нахождения решения поставленной задачи.

Пример решения

Матрица выигрышей

1 7 10
6 2 3
8 3 5
9 4 6

Доминирующих стратегий нет.

Max $Y_1 + Y_2 + Y_3$

$Y_1 + 7Y_2 + 10Y_3 \leq 1$

$6Y_1 + 2Y_2 + 3Y_3 \leq 1$

$8Y_1 + 3Y_2 + 5Y_3 \leq 1$

$9Y_1 + 4Y_2 + 6Y_3 \leq 1$

Таблица 1 – Исходная симплекс-таблица для второго игрока

| Base | Св | Y1 | Y2 | Y3 | Y4 | Y5 | Y6 | Y7 |
|------|----|----|----|----|----|----|----|----|
| F | 0 | -1 | -1 | -1 | 0 | 0 | 0 | 0 |
| 4 | 1 | 1 | 7 | 10 | 1 | 0 | 0 | 0 |
| 5 | 1 | 6 | 2 | 3 | 0 | 1 | 0 | 0 |
| 6 | 1 | 8 | 3 | 5 | 0 | 0 | 1 | 0 |
| 7 | 1 | 9 | 4 | 6 | 0 | 0 | 0 | 1 |

Ведущий столбец – Y1 (одно из минимальных значений строки индексной строки F).

Отношения свободных коэффициентов строки (берутся только положительные) к элементам ведущего столбца данной строки.

$O_2 = 1/1 = 1$

$O_3 = 1/6 = 0,16$

$O_4 = 1/8 = 0,125$

$O_5 = 1/9 = 0,11$

Ведущий элемент: $S[1][5]$

$\min(O_1, O_2, O_3) = O_1 \Rightarrow$ ведущая строка – строка с индексом 5.

При пересчете ведущей строки данная строка делится на ведущий элемент, а в ведущем столбце все элементы, кроме ведущего, заменяются 0.

В итоге получается такая таблица:

Таблица 2 – Итоговая таблица

$Y_7 = 0 - (-1) * 1/9 = 0,11$

$Y_6 = 0 + 1 * 0/9 = 0$

$Y_5 = 0 + 1 * 0/9$

$Y_{40}=0$
 $Y_{30}=-1 + 1*6/9=-0,33$
 $Y_{20}=-1+1*4/9=-0,55$
 $Y_{71}=0-1*1/9=-0,11$
 $Y_{61}=0-1*0/9=0$
 $Y_{51}=0-1*0/9=0$
 $Y_{41}=1-1*0/9=1$
 $Y_{31}=10-1*6/9=9,34$
 $Y_{21}=7-1*4/9=6,56$
 $Y_{72}=0-6*1/9=-0,66$
 $Y_{62}=0-6*0/9=0$
 $Y_{52}=1-6*0/9=1$
 $Y_{42}=0-6*0/9=0$
 $Y_{32}=3-6*6/9=-1$
 $Y_{22}=2-6*4/9=-0,66$

$Y_{73}=0-8*1/9=-0,88$
 $Y_{63}=1-8*0/9=1$
 $Y_{53}=0-8*0/9=0$
 $Y_{43}=0-8*0/9=0$
 $Y_{33}=5-8*6/9=-0,33$
 $Y_{23}=3-8*4/9=-0,55$

$CB_0=0+1*1/9=0,11$
 $CB_1=1-1*1/9=0,89$
 $CB_2=1-6*1/9=0,34$
 $CB_3=1-8*1/9=0,12$

| Base | Св | Y1 | Y2 | Y3 | Y4 | Y5 | Y6 | Y7 |
|------|------|----|-------|-------|----|----|----|-------|
| F | 0,11 | 0 | -0,55 | -0,33 | 0 | 0 | 0 | 0,11 |
| 4 | 0,89 | 0 | 6,56 | 9,34 | 1 | 0 | 0 | -0,11 |
| 5 | 0,34 | 0 | 0,66 | -1 | 0 | 1 | 0 | -0,66 |
| 6 | 0,12 | 0 | -0,55 | -0,33 | 0 | 0 | 1 | -0,88 |
| 7 | 0,11 | 1 | 0,44 | 0,66 | 0 | 0 | 0 | 0,11 |

Задача

Матрица выигрышей

2 3 1 5 6
 4 1 6 0 7
 1 2 0 0 5

Тест 3

Имеются матрицы выигрышей, содержащие экспертные оценки параметров спроектированных элементов систем защиты, предложенных разными разработчиками. Требуется получить графическое решение игры $2 \times n$ или $m \times 2$, разработав программный код с применением инструментальных средств языка Python. Результаты расчета будут использоваться для проектирования системы защиты.

В исходном файле первые два числа – размерность игры, следующие строки – матрица выигрышей игры. Программа должна прочитать из файла размерность и матрицу.

Далее для каждой строки или, соответственно, столбца вычисляются массивы $k[i]$ и $c[i]$ линейных уравнений строк/столбцов $y=kx+c$. Для этих уравнений отобразить графики черного цвета, построить график огибающей снизу/сверху синего цвета с точностью 3 знака, зеленым цветом выделить отрезок цены игры, желтым и красным отрезки p_1, p_2 (соответственно q_1, q_2). На консоль вывести цену игры и вероятности p_1, p_2 (q_1, q_2).

Пример решения

Файл

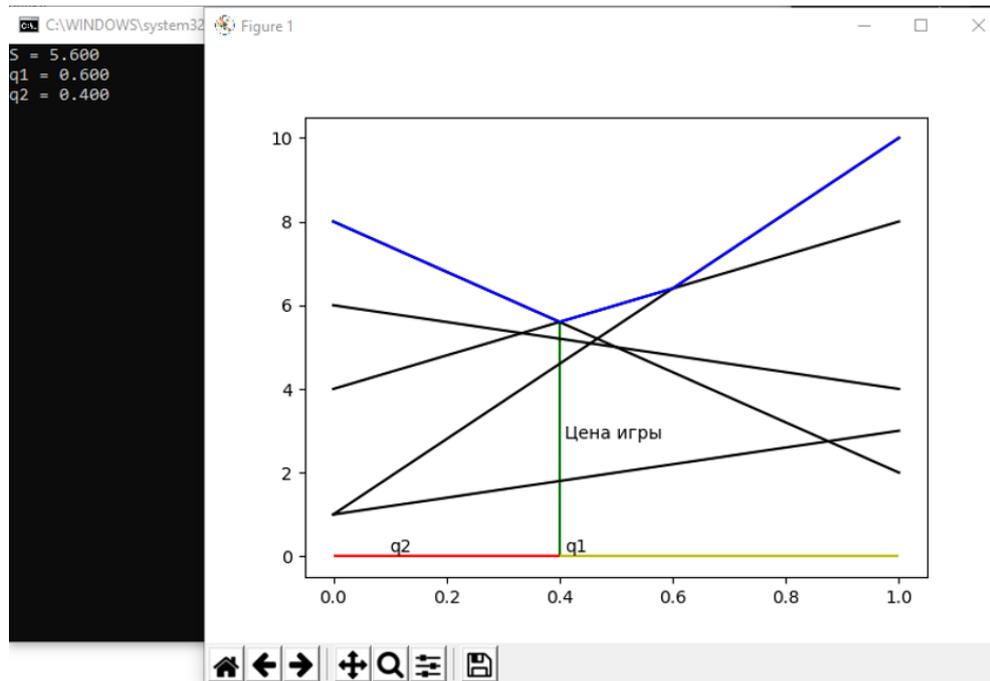
4 2

8 2

4 8

1 10

1 3



Задача

Файл

5 2

8 2

6 4

4 8

1 10

1 3

Тест 4

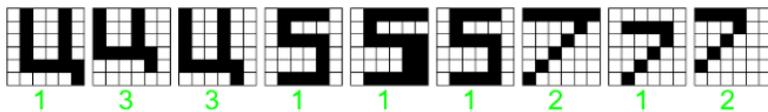
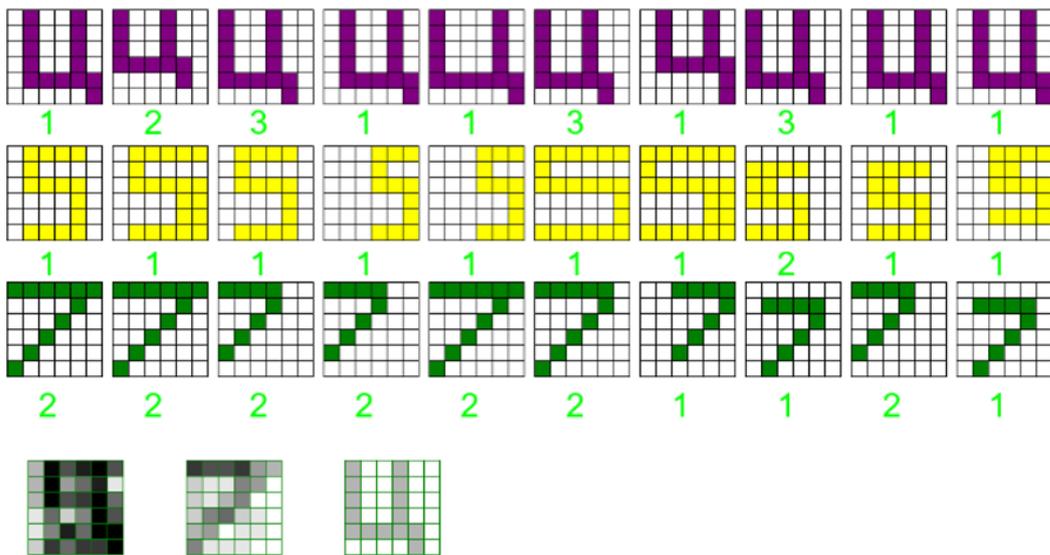
Проектирование и разработка модуля анализа битовых изображений как программного элемента системы защиты объекта информатизации. Применить инструментальные средства языка Python для разработки модуля кластерного анализа битовых изображений 6x6 методом К-внутригрупповых средних.

Создать и записать в файл 30 битовых изображений 6x6 по 10 изображений каждого из трех классов. Кроме этого, создать по 3 изображения для каждого класса и записать в другой файл. При запуске считываются изображения и выполняется кластеризация методом К-средних. После кластеризации выводятся изображения с подписью номера присвоенного класса и в виде полутоновых изображений изображения всех классов. После этого считываются тестирующие изображения, выводятся на экран с подписью к какому классу были отнесены.

Пример решения

Символы Ц 5 7

Пример решения



Задача

Выполнить кластерный анализ для символов **C F L**

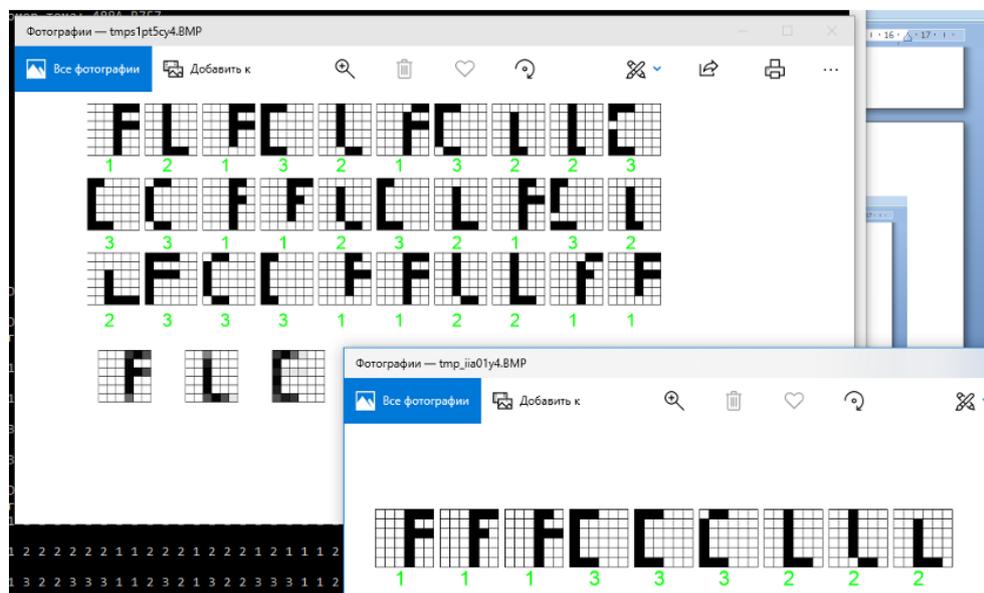
Тест 5

Проектирование и разработка модуля анализа битовых изображений как программного элемента системы защиты объекта информатизации. Применить инструментальные средства языка Python для разработки модуля кластерного анализа битовых изображений бхб методом максиминного расстояния

Создать и записать в файл 30 битовых изображений бхб по 10 изображений каждого из трех классов. Кроме этого, создать по 3 изображения для каждого класса и записать в другой файл. При запуске считываются изображения и выполняется кластеризация методом максиминного расстояния. После кластеризации выводятся изображения с подписью номера присвоенного класса и в виде полутоновых изображений изображения всех классов. После этого считываются тестирующие изображения, выводятся на экран с подписью к какому классу были отнесены.

Пример решения

Символы **F L C**



Задача

Выполнить кластерный анализ для символов **L П +**

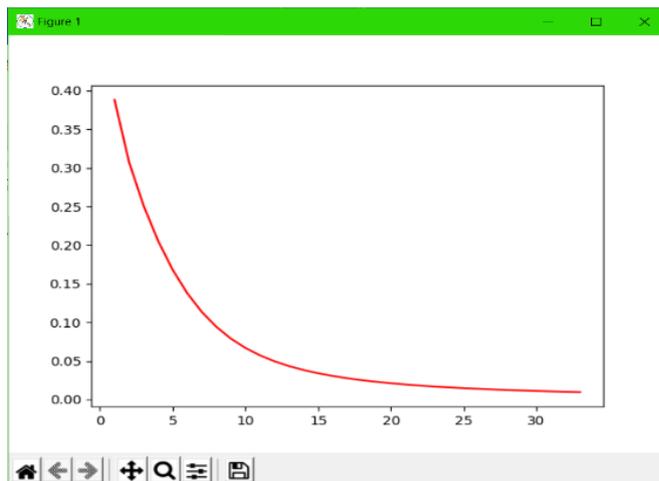
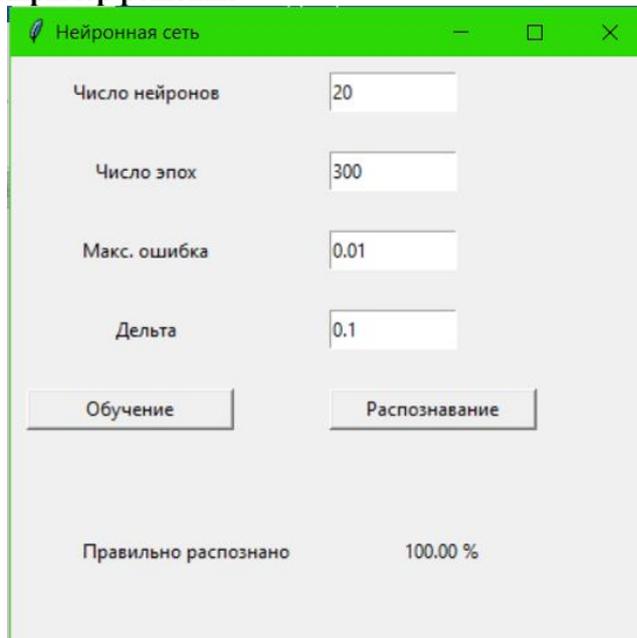
Тест 6

Проектирование и разработка модуля обучения нейронной сети для анализа битовых изображений как программного элемента системы защиты объекта информатизации. Применить инструментальные средства языка Python для разработки модуля, реализующего метод обратного распространения ошибки для обучения нейронных сетей.

Записать в файл 45 битовых изображений бхб по 15 каждого класса. Кроме этого, создать по 10 изображений для каждого класса и записать в другой файл. Для каждого образа обучающей

выборки и тестирующей указывается номер класса, к которому он относится. После тестирования выводятся изображения с подписью истинного номера изображения всех классов и активности выходного слоя. Обучение и тестирование отобразить на форме с помощью библиотеки Tkinter. График обучения отобразить с помощью matplotlib

Пример решения



Тестирование

| | | | | | | | | | | |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|---|
| | | | | | | | | | | |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0.911 | 0.831 | 0.903 | 0.800 | 0.869 | 0.892 | 0.869 | 0.897 | 0.906 | 0.831 | |
| 0.025 | 0.129 | 0.039 | 0.043 | 0.037 | 0.087 | 0.037 | 0.070 | 0.063 | 0.129 | |
| 0.098 | 0.040 | 0.174 | 0.152 | 0.107 | 0.143 | 0.107 | 0.055 | 0.039 | 0.040 | |

| | | | | | | | | | | |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|---|
| | | | | | | | | | | |
| 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| 0.135 | 0.045 | 0.107 | 0.034 | 0.078 | 0.135 | 0.040 | 0.045 | 0.049 | 0.107 | |
| 0.852 | 0.909 | 0.930 | 0.896 | 0.933 | 0.852 | 0.854 | 0.893 | 0.863 | 0.930 | |
| 0.053 | 0.131 | 0.025 | 0.172 | 0.062 | 0.053 | 0.098 | 0.098 | 0.082 | 0.025 | |

| | | | | | | | | | | |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|---|
| | | | | | | | | | | |
| 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| 0.072 | 0.168 | 0.067 | 0.114 | 0.058 | 0.071 | 0.062 | 0.058 | 0.094 | 0.233 | |
| 0.095 | 0.081 | 0.094 | 0.084 | 0.088 | 0.073 | 0.152 | 0.088 | 0.089 | 0.027 | |
| 0.812 | 0.805 | 0.878 | 0.884 | 0.850 | 0.919 | 0.804 | 0.850 | 0.828 | 0.959 | |

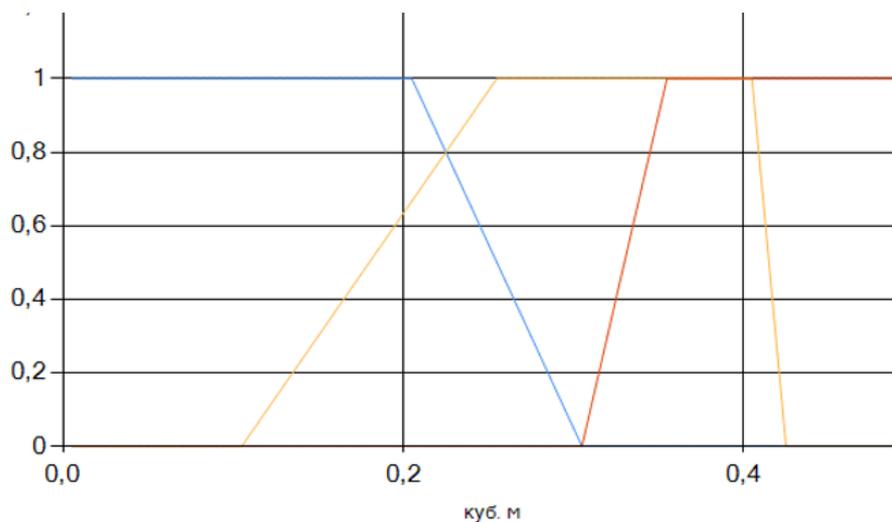
Задача

Обучить нейронную сеть для символов В П О

Тест 7

Проектирование и разработка модуля нечеткого вывода как программного элемента системы защиты объекта информатизации. Применить инструментальные средства языка Python для разработки модуля, реализующего алгоритм Мамдани для системы нечеткого вывода.

Функции принадлежности могут быть трех типов: трапециевидная, S типа и Z типа.



Треугольный тип считаем как частный случай трапецевидного.

Выполнить алгоритм Мамдани для выбранных конкретных значений .

Алгоритм содержит этапы

1. фаззификация
2. агрегирование
3. активизация
4. Аккумуляция
5. дефаззификация.

В итоге получается управляющее воздействие.

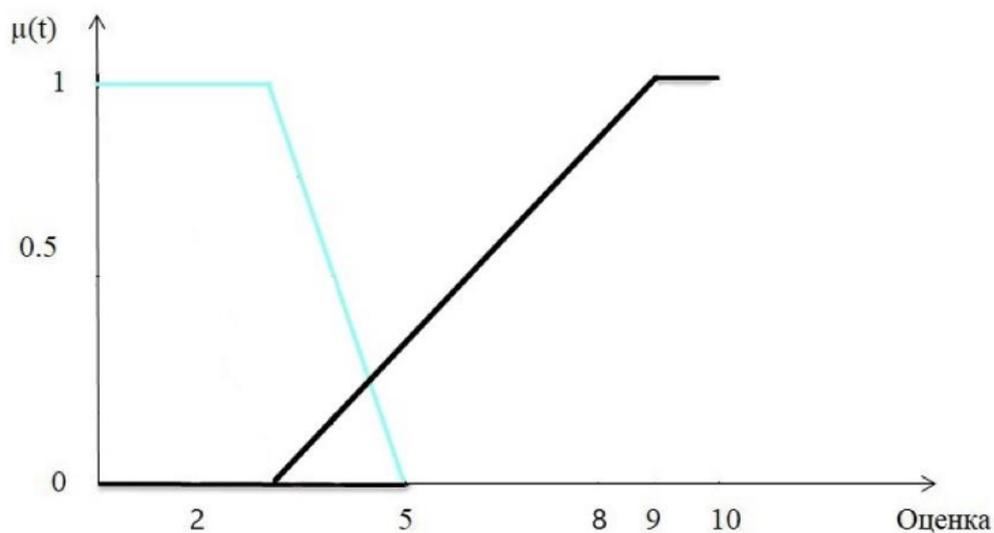
Оценка параметра Защищенность работы в интернете

Входные переменные

«Качество антивирусного программного обеспечения». Будет оцениваться, как:

хорошее,

плохое

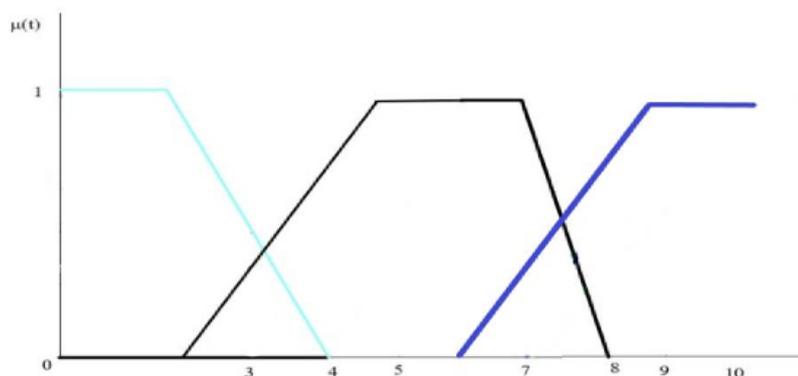


«Работа с незнакомыми сайтами», оценивается по шкале:

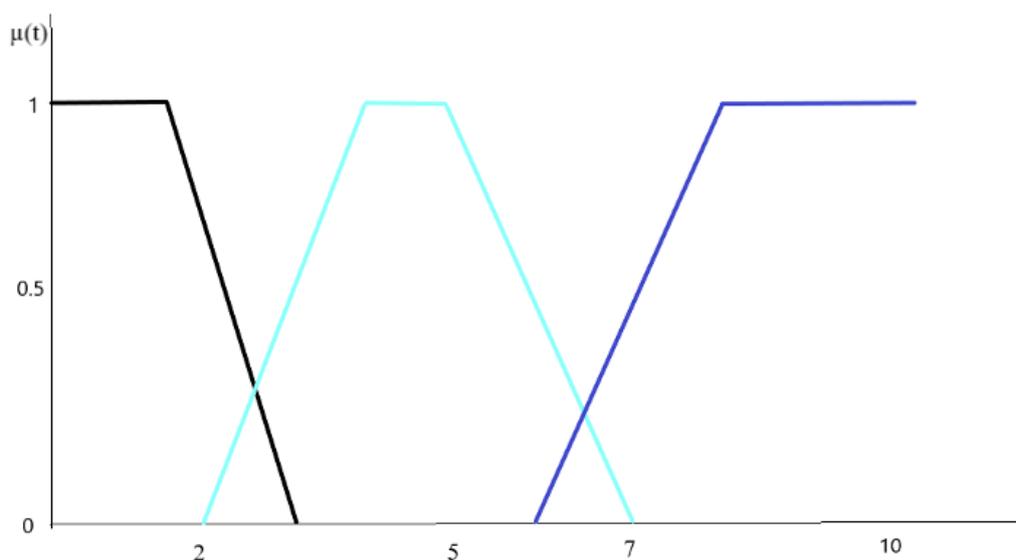
постоянная;

частая;

редкая.



Выходная переменная «Степень защиты»: низкая, средняя, высокая



Нечеткие правила

1. Если Качество плохое И Работа постоянная, ТО Степень низкая.
2. Если Качество плохое И Работа частая, ТО Степень средняя.
3. Если Качество плохое И Работа редкая, ТО Степень средняя.
4. Если Качество хорошее И Работа постоянная, ТО Степень средняя.
5. Если Качество хорошее И Работа частая, ТО Степень средняя.
6. Если Качество хорошее И Работа редкая, ТО Степень высокая.

4. *Файл и/или БТЗ с полным комплектом оценочных материалов прилагается.*