Отчёт к лабораторной работе

по дисциплине  
«Теория принятия решений»

выполнил   
студент гр. ИС/б-18-1-з Демиденко А. А.  
принял Кротов К. В.

Лабораторной работа № 6  
«Исследование применения метода анализа иерархий для решения задачи выбора альтернатив»

## Цель работы

Исследовать применение метода анализа иерархий для решения задачи выбора альтернатив.

## Программа выполнения работы

1. Сформировать следующие матрицы:
   1. парных сравнений влияния характеристик альтернатив (решений) на общую цель принятия решений (матрицу А1);
   2. парных сравнений наличия рассматриваемых свойств (характеристик) у предлагаемых к анализу решений xj – матрицы (где , m – количество критериев (свойств, характеристик) рассматриваемых альтернатив).
2. Реализовать процедуру, которая, используя заданный в варианте метод, определяет вектор собственных значений W каждой из матриц парных сравнений, вычисляет собственное значение матрицы λmax и индекс согласованности (ИС) оценок в ней.
3. Проверить выполнение условия согласованности оценок в каждой из матриц парных сравнений на каждом уровне. В случае плохой согласованности повторить шаги 1 и 2.
4. Разработать процедуру, которая на основе векторов собственных значений матриц (, количество элементов в каждом из векторов равно количеству рассматриваемых решений (альтернатив), т.е. n) для каждого из решений сформировать вектор весовых коэффициентов , каждый из которых соответствует наличию j-ой характеристики (свойства, критерия) у соответствующего i-го решения (количество элементов в каждом из этих векторов равно количеству свойств решений, влияющих на общую цель принятия решений, т.е. m).
5. Разработать процедуру, которая на основе векторов весовых коэффициентов на первом уровне – W1, на втором уровне – () выполняет расчет оценок Di для каждого решения, эта же процедура реализует определение на основе значений Di () эффективного решения .

**Вариант 1.** У студентов в процессе обучения возникает необходимость определения предмета, который они хотели бы изучать по выбору. Характеристиками (критериями), соответствующими свойствам предметов, на основе которых выполняется выбор (влияющих на выбор предмета) являются: фундаментальные знания, которые содержит преподаваемый предмет, соответствие современному уровню развития науки в данной области, возможность использования в профессиональной деятельности, симпатии к преподавателю. Для анализа и выбора могут быть предложены следующие предметы: теория принятия решений, теория алгоритмов, теория вероятностей и математическая статистика, теория информационных процессов, технологии обработки информации, технологии программирования. Для реализации выбора необходимо сформировать требуемые матрицы парных сравнений и реализовать процедуру принятия решений. При этом для определения значений элементов собственных векторов матриц парных сравнений использовать первый из предложенных в Приложении А методов.

**Метод 1**

1. Выполнить суммирование элементов каждой строки.
2. Нормализовать каждую из полученных сумм путем ее деления на сумму всех элементов.
3. Полученные результаты в сумме должны давать единицу; первый элемент результирующего вектора является приоритетом (весом) первого объекта, второй – второго и т.д.

## Ход работы

Сформируем необходимые матрицы:

type Row a = [a]

type Matrix a = [Row a]

inputA1 :: Matrix Double

inputA1 =

[ [ 1.0, 1.0, 2.0, 0.5 ] *-- фундаментальные знания, которые содержит предмет*

, [ 1.0, 1.0, 2.0, 0.5 ] *-- соответствие современному уровню развития науки*

, [ 0.5, 0.5, 1.0, 0.25 ] *-- возможность использования в проф. деятельности*

, [ 2.0, 2.0, 4.0, 1.0 ] *-- симпатии к преподавателю*

]

inputA21 :: Matrix Double *-- фундаментальные знания, которые содержит предмет*

inputA21 =

[ [ 1.0, 6.0, 4.0, 4.0, 2.0, 3.0 ] *-- теория принятия решений*

, [ 0.17, 1.0, 0.67, 0.67, 0.34, 0.5 ] *-- теория алгоритмов*

, [ 0.25, 1.5, 1.0, 1.0, 0.5, 0.75 ] *-- теория вероятностей и мат.статистика*

, [ 0.25, 1.5, 1.0, 1.0, 0.5, 0.75 ] *-- теория информационных процессов*

, [ 0.5, 3.0, 2.0, 2.0, 1.0, 1.5 ] *-- технологии обработки информации*

, [ 0.34, 2.0, 1.34, 1.34, 0.67, 1.0 ] *-- технологии программирования*

]

inputA22 :: Matrix Double *-- соответствие современному уровню развития науки в данной области*

inputA22 =

[ [ 1.0, 2.0, 1.0, 1.0, 2.0, 3.0 ] *-- теория принятия решений*

, [ 0.5, 1.0, 0.5, 0.5, 1.0, 1.5 ] *-- теория алгоритмов*

, [ 1.0, 2.0, 1.0, 1.0, 2.0, 3.0 ] *-- теория вероятностей и мат. статистика*

, [ 1.0, 2.0, 1.0, 1.0, 2.0, 3.0 ] *-- теория информационных процессов*

, [ 0.5, 1.0, 0.5, 0.5, 1.0, 1.5 ] *-- технологии обработки информации*

, [ 0.34, 0.67, 0.34, 0.34, 0.67, 1.0 ] *-- технологии программирования*

]

inputA23 :: Matrix Double *-- возможность использования в профессиональной деятельности*

inputA23 =

[ [ 1.0, 2.0, 1.0, 2.0, 2.0, 2.0 ] *-- теория принятия решений*

, [ 0.5, 1.0, 0.5, 1.0, 1.0, 1.0 ] *-- теория алгоритмов*

, [ 1.0, 2.0, 1.0, 2.0, 2.0, 2.0 ] *-- теория вероятностей и мат.статистика*

, [ 0.5, 1.0, 0.5, 1.0, 1.0, 1.0 ] *-- теория информационных процессов*

, [ 0.5, 1.0, 0.5, 1.0, 1.0, 1.0 ] *-- технологии обработки информации*

, [ 0.5, 1.0, 0.5, 1.0, 1.0, 1.0 ] *-- технологии программирования*

]

inputA24 :: Matrix Double *-- симпатии к преподавателю*

inputA24 =

[ [ 1.0, 3.0, 1.0, 1.0, 1.0, 3.0 ] *-- теория принятия решений*

, [ 0.34, 1.0, 0.34, 0.34, 0.34, 1.0 ] *-- теория алгоритмов*

, [ 1.0, 3.0, 1.0, 1.0, 1.0, 3.0 ] *-- теория вероятностей и мат.статистика*

, [ 1.0, 3.0, 1.0, 1.0, 1.0, 3.0 ] *-- теория информационных процессов*

, [ 1.0, 3.0, 1.0, 1.0, 1.0, 3.0 ] *-- технологии обработки информации*

, [ 0.34, 1.0, 0.34, 0.34, 0.34, 1.0 ] *-- технологии программирования*

]

Реализуем функционал расчёта W, λmax и ИС:

calcW :: Matrix Double -> Maybe (Row Double)

calcW matrix = if sum normalized == 1.0 then Just normalized else Nothing

where

summed = foldr ((:) . sum) [] matrix

normalized = map (/ sum summed) summed

multRow :: Num a => Row a -> Matrix a -> Row a

multRow v = map (sum . zipWith (\*) v)

calcLambda :: Matrix Double -> Maybe (Double, Double)

calcLambda matrix = do

let n = fromIntegral (length matrix)

w <- calcW matrix

let w' = multRow w matrix

let w'' = zipWith (/) w' w

let lambda = sum w'' / n

return (lambda, (lambda - n) / (n - 1))

Проверим согласованность всех матриц:

λ> calcLambda inputA1  
Just (4.0,0.0)

λ> calcLambda inputA21  
Just (6.014129955603149,2.825991120629823e-3)

λ> calcLambda inputA22  
Just (6.011626984126984,2.325396825396808e-3)

λ> calcLambda inputA23  
Just (6.0,0.0)

λ> calcLambda inputA24  
Just (6.026539682539681,5.307936507936262e-3)

Реализуем функцию формирования вектора :

import Control.Applicative (ZipList (..))

transpose :: Matrix a -> Matrix a

transpose = getZipList . traverse ZipList

getW2O :: [Matrix Double] -> Maybe (Matrix Double)

getW2O a2 = transpose <$> traverse calcW a2

Реализуем функцию для расчёта оценок Di:

calcD :: Matrix Double -> [Matrix Double] -> Maybe (Row Double)

calcD a1 a2 = do

w <- calcW a1

w2o <- getW2O a2

return $ multRow w w2o

findSolution :: Matrix Double -> [Matrix Double] -> Maybe (Double, Int)

findSolution a1 a2 = maximum <$> (zip <$> calcD a1 a2 <\*> Just [1..])

λ> findSolution inputA1 [inputA21, inputA22, inputA23, inputA24]

Just (0.26297550085538857,1)

## Выводы

В ходе работы были исследовано применение аппарата метода анализа иерархий при принятии решений по выбору альтернатив в контексте определения предмета по выбору. Для решения задачи был написан ряд функций на языке Haskell.