

Министерство образования и науки Российской Федерации
Севастопольский государственный университет
Кафедра информационных систем

Отчет
по лабораторной работе №3
“Исследование применение аппарата многомерной
полезности для принятия решений”
по дисциплине:
“ТЕОРИЯ ПРИНЯТИЯ РЕШЕНИЙ”

--	--

Выполнил: студент группы ИС/б-17-2о

Горбенко К.Н.

Проверил:

Кротов К.В.

Севастополь

2020

1 ЦЕЛЬ РАБОТЫ

Исследовать применение аппарата теории многомерной полезности при принятии решений по выбору эффективных альтернатив.

2 ПОСТАНОВКА ЗАДАЧИ

Перед выпускником учебного заведения стоит проблема выбора оптимального места дальнейшей работы. Выбор определяется значениями критериев:

К1 - величина зарплаты;

К2 - процент творческой работы;

К3 - время, за которое можно добраться до работы.

Диапазон значений для первого параметра решения (зарплата), на основании которого определяется критерий К1, заданы равным [50 тыс; 200 тыс] или в единицах тысяч – [50, 200]. Для определения многомерной функции полезности $U(k_1, k_2, k_3)$ и одномерной функции $U_1(k_1)$ на интервале [50, 200] заданы следующие значения (дискретные значения), для некоторых значения функций будут вычисляться: 50, 75, 100, 125, 150, 175, 200.

Диапазон значений параметра «процент творческой работы», на основании которых определяются значения критерия К2, задан в виде [20; 60]. Дискретные значения, для которых определяются значения функции $U(k_1, k_2, k_3)$ и одномерная функция $U_2(k_2)$ заданы следующими: 20, 30, 40, 50, 60.

Диапазон значений параметра «время, за которое можно добраться до работы», на основании которых определяются значения критерия К3, задан в виде [20; 70]. Дискретные значения, для которых определяются значения функций $U(k_1, k_2, k_3)$ и одномерная функция $U_3(k_3)$ заданы следующими: 20, 30, 40, 50, 60, 70.

Для сформированных диапазонов значений критериев необходимо определить дискретные значения одномерных функций полезности $U_1(k_1)$, $U_2(k_2)$, $U_3(k_3)$. На основании полученных значений одномерных функций полезности $U_1(k_1)$, $U_2(k_2)$, $U_3(k_3)$ должны быть определены аналитические формы этих функций (для постановки в них произвольных значений рассматриваемых параметров,

характеризующих решения). Для них должны быть заданы следующие аналитические формы:

$$U_1(k_1) = a_1 k_1 + b_1 (k_1)^2,$$

$$U_2(k_2) = a_2 k_2 + b_2 (k_2)^2,$$

$$U_3(k_3) = a_3 k_3 + b_3 (k_3)^2.$$

Для определения коэффициентов в приведенных аналитических функциях $U_1(k_1)$, $U_2(k_2)$, $U_3(k_3)$ необходимо применить метод наименьших квадратов. Т.к. аналитические формы выражений для $U_1(k_1)$ и $U_2(k_2)$ получены, а многомерная полезность $U(k_1, k_2, k_3)$ является аддитивной функцией, тогда для заданных в таблице значений параметров определить эффективное решение.

3 КОД ПРОГРАММЫ

```
import numpy as np
import matplotlib.pyplot as plt
import pprint
from pylab import *
from scipy.linalg import *
from operator import itemgetter

class SquareFunction:
    def __init__(self, a, b, c):
        self.a = a
        self.b = b
        self.c = c

    def get_value(self, x):
        return self.a * (x**2) + self.b * x + self.c

pp = pprint.PrettyPrinter(indent=4)
file = input('Input filename: ')
input_file = open(file + '.input', 'r')
criterias_count = int(input_file.readline())
criterias = []
for c in range(criterias_count):
    criteria = {}
    criteria['name'] = input_file.readline().strip()
    criteria['values'] = list(map(lambda x: float(x), input_file.readline().split(';')))
    if input_file.readline().strip() == 'inverted':
```

```

        criteria['values'] = list(map(lambda x: 1 / x, criteria['values']))
        criteria['inverted'] = True
    else:
        criteria['inverted'] = False

    # calculate criteria profitability function
    subplot = plt.subplot(3, 1, c + 1)
    x = np.array(criteria['values'])
    y = np.array([i for i in range(len(x))])
    m = vstack((x**2, x, ones(len(x)))).T
    a, b, k = lstsq(m, y)[0]
    x_s = linspace(x[0], x[-1])
    subplot.title.set_text("%s: %10.4f*x^2+%10.4f*x+%10.4f" % (criteria['name'], a, b, k))
    subplot.plot(x_s, a*(x_s**2) + b*x_s + k, '--')
    subplot.plot(x, y, '.', marker='o', markersize = 4)
    criteria['function'] = SquareFunction(a, b, k)

    criterias.append(criteria)

plt.show()

alternatives_count = int(input_file.readline())
alternatives = []
header = list(map(lambda x: x.strip(), input_file.readline().split(';')))

for i in range(alternatives_count):
    line = list(map(lambda x: x.strip(), input_file.readline().split(';')))
    alternative = {}
    alternative['name'] = line[0]
    profitability = 0
    for c in range(criterias_count):
        val = float(line[c + 1])
        if criterias[c]['inverted']:
            val = 1 / val
        alternative[header[c + 1]] = val
        profitability += criterias[c]['function'].get_value(val)
    alternative['profitability'] = profitability
    alternatives.append(alternative)

pp.pprint(criterias)
pp.pprint(alternatives)

pp.pprint(max(alternatives, key=itemgetter('profitability')))
```

4 РЕЗУЛЬТАТ ВЫПОЛНЕНИЯ ПРОГРАММЫ

Входные данные	Выходные данные
<pre> 7 0 0 0 0 0 0 0 1 0 1 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 1 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 1 0 0 1 0 0 0 0 1 0 0 1 0 1 0 0 1 0 0 1 0 1 0 0 1 0 0 0 0 1 0 0 1 </pre>	<pre> {1: -1, 3: -1, 6: -1, 2: 0, 5: 0, 4: 1, 7: 1} {frozenset({1, 3, 6}): -1, frozenset({2, 5}): 0, frozenset({4, 7}): 1} frozenset({4, 7}) </pre>

ВЫВОДЫ

В ходе лабораторной работы изучен способ применения аппарата теории полезности при принятии решений по выбору альтернатив. Использование функции полезности предполагает определение числовых значений, характеризующих решения, связанные отношением предпочтения. При этом более предпочтительным решениям соответствуют большие значения функции полезности ($x_i \succ x_j \Leftrightarrow U(x_i) > U(x_j)$). В этом случае, отношение предпочтения является транзитивным по полезности.

По заданию была разработана программа, формирующая множество классов эквивалентности по заданным матрицам отношений строгого предпочтения и эквивалентности. Полученные классы эквивалентности упорядочиваются в смысле предпочтительности решений одного класса решениями другого и для классов определяются соответствующие значения функции полезности. Решения класса с наибольшим значением функции полезности являются предпочтительными.