Министерство образования и науки Российской Федерации

Севастопольский государственный университет

Институт информационных технологий

Кафедра ИС

# ОТЧЕТ

по лабораторной работе №3

ИССЛЕДОВАНИЕ ПРИМЕНЕНИЕ АППАРАТА МНОГОМЕРНОЙ ПОЛЕЗНОСТИ ДЛЯ ПРИНЯТИЯ РЕШЕНИЙ

Выполнил:

ст. гр. ИС/б-21-2-о

Мовенко К. М.

Проверил:

Кротов К.В.

Севастополь

2024

# ЦЕЛЬ РАБОТЫ

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

# ЗАДАНИЕ

Вариант 3. Перед ЛПР стоит проблема выбора объекта недвижимости, в который он может вложить средства (покупка дачи). Выбор определяется значением критериев:

− качество дачи;

− расстояние до города;

− цена.

Диапазон значений параметра «качество дачи», на основании которых определяются значения критерия , задан равным [20;100] (измеряется в процентах). Для определения многомерной функции полезности и одномерной функции на интервале [20;100] заданы следующие дискретные значения: 20, 40, 60, 80, 100.

Диапазон значений параметра «расстояние до города», на основании которых определяются значения критерия , задан равным [20;120]. Дискретные значения, для которых определяются значения функций и одномерной функции , заданы следующими: 20, 40, 60, 80, 100, 120 (чем расстояние до города ниже, тем полезность больше).

Диапазон значений параметра «цена», на основании которых определяются значения критерия , задан в виде [20;70]. Дискретные значения, для которых определяются значения функций и одномерной функции , заданы следующими: 20, 30, 40, 50, 60, 70 (чем цена ниже, тем полезность больше).

Для сформированных диапазонов значений критериев необходимо определить дискретные значения одномерных функций полезности , , .

На основании полученных значений одномерных функций полезности , , должны быть определены аналитические формы этих функций (для подстановки в них произвольных значений рассматриваемых параметров, характеризующих решения). Для них должны быть заданы следующие аналитические формы: , , . Для определения коэффициентов в приведенных аналитических функциях необходимо применить метод наименьших квадратов.

На основе полученных результатов для заданных в таблице значений параметров определить эффективное решение.



Рисунок 1 – Таблица решений и их критериев

# ХОД РАБОТЫ

Была написана программа, производящая все необходимые вычисления. В первую очередь она сопоставляет дискретные значения параметров с ожидаемыми для них значениями полезности, а затем по полученным точкам находит параметры соответствующих одномерных функций полезности.

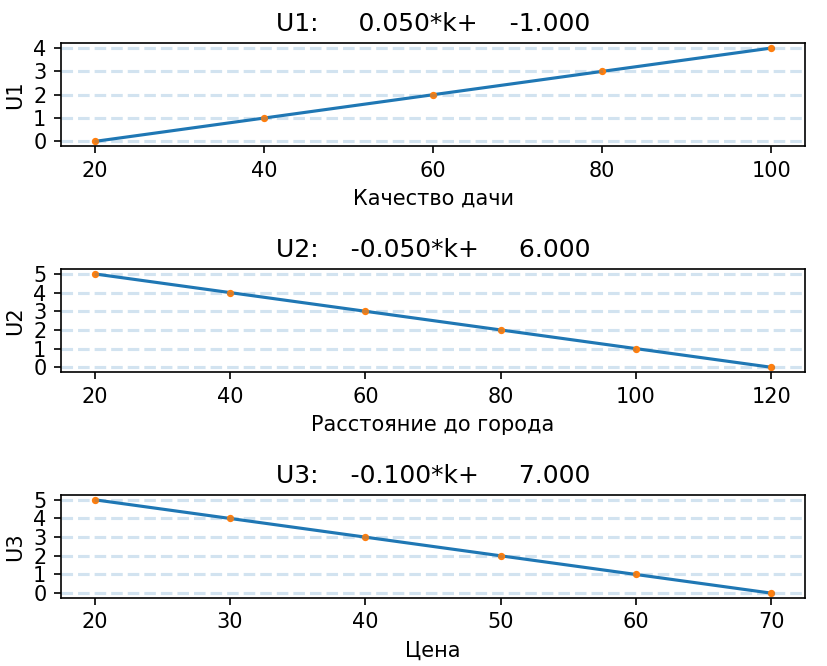


Рисунок 2 – Графики одномерных функций полезности для критериев

Получив одномерные функции полезности, программа вычисляет для каждого решения полезность соответствующих критериев, а затем агрегирует их, получая тем самым значение трёхмерной функции полезности. При агрегировании учитывается аддитивность структуры предпочтений. Наибольшей общей полезностью обладает решение .

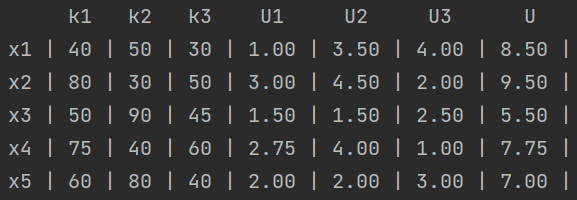


Рисунок 3 – Значения одномерных и трёхмерной функции полезности

# ТЕКСТ ПРОГРАММЫ

import numpy as np  
import matplotlib.pyplot as plt  
  
  
*# значение аддитивной функции полезности по трём критериям*def findU(k1, k2, k3):  
 u1 = a[0] \* k1 + b[0]  
 u2 = a[1] \* k2 + b[1]  
 u3 = a[2] \* k3 + b[2]  
  
 return u1 + u2 + u3  
  
  
*# множество решений*X = [(40, 50, 30),  
 (80, 30, 50),  
 (50, 90, 45),  
 (75, 40, 60),  
 (60, 80, 40)]  
  
*# дискретные значения критериев*K1 = [20, 40, 60, 80, 100]  
K2 = [20, 40, 60, 80, 100, 120]  
K3 = [20, 30, 40, 50, 60, 70]  
  
K = [K1, K2, K3] *# множество критериев*inverted = [False, True, True] *# "обратность" критериев*n = len(X) *# число решений*m = int(len(K)) *# число критериев*crit\_names = ['Качество дачи', 'Расстояние до города', 'Цена']  
  
a, b = list(), list() *# значения a, b для Ui=ai\*ki+bi  
  
# вычисление коэффициентов одномерных Ui и вывод графиков*for i in range(len(K)):  
 x = np.array(K[i].copy())  
 y = np.array([i for i in range(len(K[i]))])  
  
 if inverted[i]:  
 x = x[::-1]  
  
 subplot = plt.subplot(3, 1, i+1)  
  
 sum\_x = sum(x)  
 sum\_y = sum(y)  
 sum\_xy = sum(x \* y)  
 sum\_x\_sq = sum(x \*\* 2)  
  
 *# метод наименьших квадратов* a\_i = (len(x) \* sum\_xy - sum\_x \* sum\_y) / (len(x) \* sum\_x\_sq - (sum\_x \*\* 2))  
 b\_i = (sum\_y - a\_i \* sum\_x) / len(x)  
  
 a.append(a\_i)  
 b.append(b\_i)  
  
 x\_s = np.linspace(x[0], x[-1])  
  
 *# формирование подграфика* subplot.title.set\_text("U%i:%10.3f\*k+%10.3f" % (i+1, a\_i, b\_i))  
 subplot.plot(x\_s, a\_i \* x\_s + b\_i)  
 subplot.plot(x, y, '.', markersize=5)  
  
 subplot.set\_xlabel(crit\_names[i])  
 subplot.set\_ylabel('U%i' % (i+1))  
  
 for y\_i in y:  
 subplot.axhline(y=y\_i, linestyle='--', alpha=0.2)  
  
 subplot.set\_xticks(x)  
 subplot.set\_yticks(y)  
  
plt.subplots\_adjust(hspace=1.2)  
plt.show()  
  
Uk = [[] for \_ in range(m)] *# одномерные полезности*U = [0 for \_ in range(n)] *# многомерная полезность  
  
# значения многомерной полезности*for i in range(n):  
 for j in range(m):  
 result = a[j] \* X[i][j] + b[j]  
 Uk[j].append(result)  
 U[i] += result  
  
  
*# ВЫВОД РЕЗУЛЬТАТОВ  
  
# входные данные: решения и частные критерии*print(' k1 k2 k3 U1 U2 U3 U ')  
for i in range(n):  
 print('x%i | %2i | %2i | %2i | %.2f | %.2f | %.2f | %.2f |' %  
 (i+1, X[i][0], X[i][1], X[i][2], Uk[0][i], Uk[1][i], Uk[2][i], U[i]))  
print()  
  
*# матрицы предпочтения*for k3 in K3:  
 print('k3 = %i' % k3)  
 for k2 in K2:  
 for k1 in K1:  
 print('%2.0f' % findU(k1, k2, k3), end=' ')  
 print()  
 print()

# ВЫВОД

В ходе работы было исследовано применение аппарата теории многомерной полезности при принятии решений по выбору эффективных альтернатив