

Міністерство освіти і науки України
Національний технічний університет України «КПІ» імені Ігоря Сікорського
Кафедра обчислювальної техніки ФІОТ

ЗВІТ
з лабораторної роботи №1
з навчальної дисципліни «Методи Оптимізації та Планування Експерименту»

Виконав:
Студент 2 курсу кафедри ОТ
ФІОТ,
Навчальної групи ІО-93
Верцанов С. С.

Перевірив:
Асистент кафедри ОТ ФІОТ
Регіда П. Г.

Київ 2021

Завдання (варіант №305):

- 1) Використовуючи програму генерації випадкових чисел, провести трьохфакторний експеримент в восьми точках (три стовбці і вісім рядків в матриці планування – заповнити її випадковими числами). Рекомендовано взяти обмеження до 20 при генерації випадкових чисел, але врахувати можливість зміни обмеження на вимогу викладача. Програма створюється на основі будь-якої мови високого рівня.
- 2) Визначити значення функції відгуків для кожної точки плану за формулою лінійної регресії: $Y = a_0 + a_1X_1 + a_2X_2 + a_3X_3$, де a_0, a_1, a_2, a_3 довільно вибрані (для кожного студента різні) коефіцієнти, постійні протягом усього часу проведення експерименту.
- 3) Виконати нормування факторів. Визначити значення нульових рівнів факторів. Знайти значення відгуку для нульових рівнів факторів і прийняти його за еталонне $Y_{\text{ет}}$.
- 4) Знайти точку плану, що задовольняє критерію вибору оптимальності. Варіанти обираються по номеру в списку в журналі викладача.

305	$\max((Y - Y_{\text{ет}})^2)$
-----	-------------------------------

Код та результат виконання:

```
import random as rand
from prettytable import PrettyTable

a0 = 5
a1 = 12
a2 = 8
a3 = 11

experiment_matrix = [['№', 'X1', 'X2', 'X3', 'Y', 'Xn1', 'Xn2', 'Xn3']]
X1_arr = []
X2_arr = []
X3_arr = []
Y_arr = []

for i in range(8):
    X1_arr.append(rand.randint(1, 20))
    X2_arr.append(rand.randint(1, 20))
    X3_arr.append(rand.randint(1, 20))
    Y_arr.append(a0 + a1 * X1_arr[i] + a2 * X2_arr[i] + a3 * X3_arr[i])
    experiment_matrix.append([i+1, X1_arr[i], X2_arr[i], X3_arr[i], Y_arr[i]])

x0_arr = ['x0', (max(X1_arr) + min(X1_arr)) / 2, (max(X2_arr) + min(X2_arr)) / 2,
(max(X3_arr) + min(X3_arr)) / 2]
Y_et = a0 + a1 * x0_arr[1] + a2 * x0_arr[2] + a3 * x0_arr[3]
x0_arr.append(Y_et)
for i in range(3):
    x0_arr.append(' ')

experiment_matrix.append(x0_arr)

dx_arr = ['dx', x0_arr[1] - min(X1_arr), x0_arr[2] - min(X2_arr), x0_arr[3] -
min(X3_arr)]
```

```

experiment_matrix.append(dx_arr)
for i in range(4):
    dx_arr.append(' ')
Xn1_arr = []
Xn2_arr = []
Xn3_arr = []

for i in range(8):
    experiment_matrix[i + 1].append(round((X1_arr[i] - x0_arr[1])/dx_arr[1], 3))
    experiment_matrix[i + 1].append(round((X2_arr[i] - x0_arr[2]) / dx_arr[2], 3))
    experiment_matrix[i + 1].append(round((X3_arr[i] - x0_arr[3]) / dx_arr[3], 3))

choice_Y = []
for i in range(len(Y_arr)):
    choice_Y.append([(Y_arr[i]-Y_et)**2, i])

chosen_y = max(choice_Y, key=lambda x: x[0])[1]

experiment_table = PrettyTable()
experiment_table.field_names = experiment_matrix[0]
for i in range(1, len(experiment_matrix)):
    experiment_table.add_row(experiment_matrix[i])
print(experiment_table)
print("Criterion for selecting optimality's arguments is №{} : {}, {}, {}"
      .format(chosen_y + 1, X1_arr[chosen_y],
X2_arr[chosen_y], X3_arr[chosen_y]))

```

C:\Users\Vercabra\AppData\Local\Programs\Python\Python39\python.exe
"D:/Labas/МОПЕ/Laba 1/Task.py"

№	X1	X2	X3	Y	Xn1	Xn2	Xn3
1	4	5	15	258	-1.0	-0.467	1.0
2	19	4	11	386	1.0	-0.6	0.333
3	18	10	8	389	0.867	0.2	-0.167
4	10	5	9	264	-0.2	-0.467	0.0
5	10	11	4	257	-0.2	0.333	-0.833
6	9	1	5	176	-0.333	-1.0	-0.667
7	5	14	3	210	-0.867	0.733	-1.0
8	9	16	11	362	-0.333	1.0	0.333
x0	11.5	8.5	9.0	310.0			
dx	7.5	7.5	6.0				

Criterion for selecting optimality's arguments is №6 : 9, 1, 5

Process finished with exit code 0