

Національний технічний університет України
«Київський політехнічний інститут ім. І. Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки

Методи оптимізації та планування експерименту

Лабораторна робота №1

«Загальні принципи організації експериментів з довільними значеннями факторів»

Виконав:

студент групи ІО-93

Варченко Є. В.

Номер у списку групи – 3

Перевірив:

ас. Регіда П. Г.

Київ – 2021

Лабораторна робота №1

Тема: «Загальні принципи організації експериментів з довільними значеннями факторів».

Мета: вивчити основні поняття, визначення, принципи теорії планування експерименту, на основі яких вивчити побудову формалізованих алгоритмів проведення експерименту і отримання формалізованої моделі об'єкта. Закріпити отримані знання практичним їх використанням при написанні програми, що реалізує завдання на лабораторну роботу.

Завдання

- 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 відгукув для кожної точки плану за формулою лінійної регресії:
$$Y = a_0 + a_1 X_1 + a_2 X_2 + a_3 X_3,$$
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 $Y_{\text{эт}}$.
- 4) Знайти точку плану, що задовольняє критерію вибору оптимальності (див. табл.1). Варіанти обираються по номеру в списку в журналі викладача.
- 5) Скласти вираз для функції відгуку, підставивши замість X_i значення факторів в точці, що задовольняє критерію вибору.

303	$\max(Y)$
-----	-----------

Код програми

```
import kotlin.random.Random

fun x0Find(X: IntArray): Float {
    return (X.maxOrNull()!! - X.minOrNull()!!) / 2.0f
}

fun dxFind(x0: Float, X: IntArray): Float {
    return X.maxOrNull()!! - x0
}

fun xNormArrayFind(x0: Float, dx: Float, X: IntArray): FloatArray {
    return FloatArray(8) { (X[it] - x0) / dx }
}

fun YFind(a0: Int, a1: Int, a2: Int, a3: Int, X1: Float, X2: Float, X3: Float):
Float {
    return a0 + a1 * X1 + a2 * X2 + a3 * X3
}

fun maxYFind(Y: FloatArray): Float {
    return Y.maxOrNull()!!
}

fun main() {
    val min = 0
    val max = 50
}
```

```
val a0: Int = Random.nextInt(min, max)
val a1: Int = Random.nextInt(min, max)
val a2: Int = Random.nextInt(min, max)
val a3: Int = Random.nextInt(min, max)
```

```
val X1 = IntArray(8) { Random.nextInt(min, max) }
val X2 = IntArray(8) { Random.nextInt(min, max) }
val X3 = IntArray(8) { Random.nextInt(min, max) }
```

```
val Y = FloatArray(8) { YFind(a0, a1, a2, a3, X1[it].toFloat(),
X2[it].toFloat(), X3[it].toFloat()) }
```

```
val x01 = x0Find(X1)
val x02 = x0Find(X2)
val x03 = x0Find(X3)
```

```
val dx1 = dxFind(x01, X1)
val dx2 = dxFind(x02, X2)
val dx3 = dxFind(x03, X3)
```

```
val Xn1 = xNormArrayFind(x01, dx1, X1)
val Xn2 = xNormArrayFind(x02, dx2, X2)
val Xn3 = xNormArrayFind(x03, dx3, X3)
```

```
val Yet = YFind(a0, a1, a2, a3, x01, x02, x03)
```

```
val optimalPointOfX1 = X1[Y.indexOfFirst { it == maxYFind(Y) }]
val optimalPointOfX2 = X2[Y.indexOfFirst { it == maxYFind(Y) }]
val optimalPointOfX3 = X3[Y.indexOfFirst { it == maxYFind(Y) }]
```

```
println("${"№".padStart(2)} | " +
        "${"X1".padEnd(5)} | " +
        "${"X2".padEnd(5)} | " +
        "${"X3".padEnd(5)} | " +
        "${"Y".padEnd(7)} | " +
```

```

    "${Xn1".padEnd(9)} | " +
    "${Xn2".padEnd(9)} | " +
    "Xn3")

```

```
println("-".repeat(78))
```

```

for (i in 0 until 8) {
    println("${(i + 1).toString().padStart(2)} | " +
        "${X1[i].toString().padEnd(5)} | " +
        "${X2[i].toString().padEnd(5)} | " +
        "${X3[i].toString().padEnd(5)} | " +
        "${Y[i].toString().padEnd(7)} | " +
        "${String.format("%.4f", Xn1[i]).padEnd(9)} | " +
        "${String.format("%.4f", Xn2[i]).padEnd(9)} | " +
        String.format("%.4f", Xn3[i])
    )
}

```

```

println("${x0".padEnd(2)} | " +
    "${x01.toString().padEnd(5)} | " +
    "${x02.toString().padEnd(5)} | " +
    "${x03.toString().padEnd(5)} | ")

```

```

println("${dx".padEnd(2)} | " +
    "${dx1.toString().padEnd(5)} | " +
    "${dx2.toString().padEnd(5)} | " +
    "${dx3.toString().padEnd(5)} | ")

```

```
println("\nYet = $Yet")
```

```

println("Selection criterion: max(Y). Optimal point: ($optimalPointOfX1,
$optimalPointOfX2, $optimalPointOfX3)")
}

```

Результати роботи програми

a_0, a_1, a_2, a_3 – випадково згенеровані:

Nº	X1	X2	X3	Y	Xn1	Xn2	Xn3
1	16	17	43	1815.0	-0.0857	-0.2340	1.0000
2	12	39	7	1447.0	-0.3143	0.7021	-0.4400
3	8	42	33	2245.0	-0.5429	0.8298	0.6000
4	7	32	11	1324.0	-0.6000	0.4043	-0.2800
5	22	8	34	1323.0	0.2571	-0.6170	0.6400
6	8	1	9	343.0	-0.5429	-0.9149	-0.3600
7	0	46	28	2185.0	-1.0000	1.0000	0.4000
8	35	27	8	1230.0	1.0000	0.1915	-0.4000
x0	17.5	22.5	18.0				
dx	17.5	23.5	25.0				

Yet = 1287.5

Selection criterion: max(Y). Optimal point: (8, 42, 33)

$a_0 = 3, a_1 = 5, a_2 = 7, a_3 = 10$:

№	X1	X2	X3	Y	Xn1	Xn2	Xn3
1	27	41	23	655.0	0.4194	1.0000	0.0370
2	45	6	40	670.0	1.0000	-0.5556	0.6667
3	17	37	36	707.0	0.0968	0.8222	0.5185
4	43	32	49	932.0	0.9355	0.6000	1.0000
5	25	28	6	384.0	0.3548	0.4222	-0.5926
6	24	4	42	571.0	0.3226	-0.6444	0.7407
7	44	6	5	315.0	0.9677	-0.5556	-0.6296
8	23	15	24	463.0	0.2903	-0.1556	0.0741
x0	14.0	18.5	22.0				
dx	31.0	22.5	27.0				

Yet = 422.5
Selection criterion: max(Y). Optimal point: (43, 32, 49)

Висновки

- Ознайомилися з темою роботи.
- Були здобуті необхідні навички для виконання завдань.
- Розроблено програму, яка виконує поставлену задачу.
- Вище приведені результати свідчать про успішне виконання умов завдань.
- Основну мету лабораторної роботи було досягнуто.