**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ**

**“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМ. ІГОРЯ СІКОРСЬКОГО”**

**Л А Б О Р А Т О Р Н А Р О Б О Т А № 2**

***ПРОВЕДЕННЯ ДВОФАКТОРНОГО ЕКСПЕРИМЕНТУ З ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ***

**З дисципліни:**

**«Теорія планування експерименту»**

Виконали:

Студенти ФІОТ

Бригада #3

Групи ІО – 53

Венгерак Арсеній,

Возний Олександр

**Київ 2017**

**Лабораторна робота №2**

ПРОВЕДЕННЯ ДВОФАКТОРНОГО ЕКСПЕРИМЕНТУ З ВИКОРИСТАННЯМ ЛІНІЙНОГО РІВНЯННЯ РЕГРЕСІЇ

**Мета завдання:** провести двофакторний експеримент, перевірити однорідність дисперсії за критерієм Романовського, отримати коефіцієнти рівняння регресії, провести натуралізацію рівняння регресії.

Варіант завдання: 103

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | X1min |  |  | X1max | X2min |  | X2max | Ymin | Ymax |
| 103 | -20 |  |  | 30 | 30 |  | 80 | (20-103)\*10= -830 | (30-103)\*10= -730 |

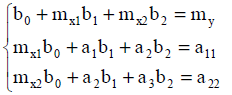
**Підготовчі дані.**

Нормована матриця планування експерименту:

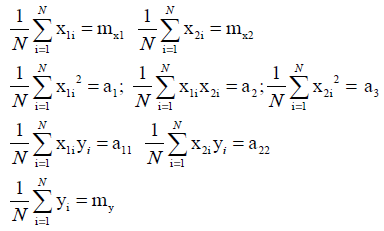


**Хід роботи:**

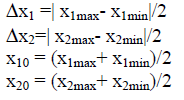
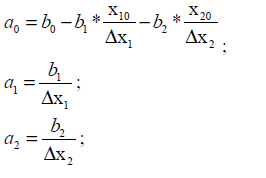
1. Складаємо матрицю планування експерименту.
2. Проводимо експеримент в усіх точках плану.
3. Перевіряємо однорідність дисперсії за критерієм Романовського. Якщо дисперсії однорідні, то проводимо розрахунок коефіцієнтів рівняння регресії. Якщо дисперсії неоднорідні, то необхідно збільшити m – кількість дослідів y за однієї і тієї ж комбінації факторів, провести нові досліди і перевірити критерій знову.
4. Обчислюємо нормовані коефіцієнти b0, b1 , b2рівняння регресії **y = b0 + b1\*x1 + b2\* x2**, знаходимо їх з системи рівнянь методом Крамера:

****

Де введено такі позначення:

****

5. Обчислюємо натуралізовані коефіцієнти рівняння регресії **y = a0 + a1\*x1 + a2\* x2**.

** **

**Результат виконання експерименту:**

-768.3527674984655 -778.0912735761959 -792.1844018536144 -759.9775720534573 -811.0903717160597 -752.8632283004297

-738.6984519148225 -822.860498944323 -816.8595169366782 -803.4307848724517 -772.660202792459 -763.2329384639295

-789.8245896355863 -736.3473287290885 -752.1375313008034 -755.7897932990696 -745.0466302965647 -748.7524248501472

Середнє у1 = -777.0932691663705

Середнє у2 = -786.290398987444

Середнє у3 = -754.6497163518767

Сигма у1 = 390.16326963215306

Сигма у2 = 924.6763619346178

Сигма у3 = 284.4762102063792

Основне відхилення тета0 = 1.2909944487358056

Fuv1 = 2.3699728649660052

Fuv2 = 1.371514578843345

Fuv3 = 3.2504523357640065

Тета uv1 = 1.5799819099773367

Тета uv2 = 0.9143430525622299

Тета uv3 = 2.166968223842671

Ruv1 = 0.44925205568875887

Ruv2 = 0.06634958618268952

Ruv3 = 0.9039296992991827

Ruv < Rkp, дисперсія однорідна!

b0 = -770.4700576696602

b1 = 11.22177640724696

b2 = -4.5985649105366475

Нормоване рівняння регресії:

y = (-770.4700576696602) + (11.22177640724696) \* X1 + (-4.5985649105366475) \* X2

Перевірка:

-777.0932691663704 = -777.0932691663705

-786.2903989874437 = -786.290398987444

-754.6497163518766 = -754.6497163518767

a0 = -762.597570147929

a1 = 0.4488710562898784

a2 = -0.1839425964214659

Натуралізоване рівняння регресії:

y = (-762.597570147929) + (0.4488710562898784) \* X1 + (-0.1839425964214659) \* X2

Перевірка:

-777.0932691663705 = -777.0932691663705

-786.2903989874438 = -786.290398987444

-754.6497163518766 = -754.6497163518767

Process finished with exit code 0

**Лістинг:**

**Experiment.java**

**package com;**

**/\*\***

**\* Created by User on 19.10.2016.**

**\*/**

**public class Experiment {**

**private static double R\_critical = 2.10; //number = 6, p = 0.95**

**private static double X11 = -1.0;**

**private static double X12 = -1.0;**

**private static double X13 = +1.0;**

**private static double X21 = -1.0;**

**private static double X22 = +1.0;**

**private static double X23 = -1.0;**

**private int minX1;**

**private int maxX1;**

**private int minX2;**

**private int maxX2;**

**private int minY;**

**private int maxY;**

**private double[] averageY;**

**private double b0;**

**private double b1;**

**private double b2;**

**private int numberOfY;**

**private double[][] Y;**

**public Experiment(int minX1, int maxX1, int minX2, int maxX2, int minY, int maxY, int numberOfY) {**

**this.minX1 = minX1;**

**this.maxX1 = maxX1;**

**this.minX2 = minX2;**

**this.maxX2 = maxX2;**

**this.minY = minY;**

**this.maxY = maxY;**

**this.numberOfY = numberOfY;**

**randomY();**

**romanovskiy();**

**normalize();**

**naturalize();**

**}**

**private void randomY() {**

**Y = new double[3][numberOfY];**

**for (int i = 0; i < Y.length; i++) {**

**for (int j = 0; j < Y[i].length; j++) {**

**Y[i][j] = Math.random() \* (maxY - minY) + minY;**

**}**

**}**

**for (int i = 0; i < 3; i++) {**

**for (int j = 0; j < numberOfY; j++) {**

**System.out.print(Y[i][j]+ " ");**

**}**

**System.out.println();**

**}**

**System.out.println();**

**}**

**private void romanovskiy() {**

**averageY = new double[3];**

**for (int i = 0; i < numberOfY; i++) {**

**averageY[0] += Y[0][i];**

**averageY[1] += Y[1][i];**

**averageY[2] += Y[2][i];**

**}**

**averageY[0] /= numberOfY;**

**averageY[1] /= numberOfY;**

**averageY[2] /= numberOfY;**

**System.out.println("Середнє у1 = " + averageY[0]);**

**System.out.println("Середнє у2 = " + averageY[1]);**

**System.out.println("Середнє у3 = " + averageY[2]);**

**System.out.println();**

**double sigmaY1 = 0;**

**double sigmaY2 = 0;**

**double sigmaY3 = 0;**

**for (int i = 0; i < numberOfY; i++) {**

**sigmaY1 += Math.pow(Y[0][i] - averageY[0], 2);**

**sigmaY2 += Math.pow(Y[1][i] - averageY[1], 2);**

**sigmaY3 += Math.pow(Y[2][i] - averageY[2], 2);**

**}**

**sigmaY1 /= numberOfY;**

**sigmaY2 /= numberOfY;**

**sigmaY3 /= numberOfY;**

**System.out.println("Сигма у1 = " + sigmaY1);**

**System.out.println("Сигма у2 = " + sigmaY2);**

**System.out.println("Сигма у3 = " + sigmaY3);**

**System.out.println();**

**double teta0 = Math.pow(2.0 \* (2.0 \* (double)numberOfY - 2.0) / (double)numberOfY / ((double)numberOfY - 4.0), 0.5);**

**System.out.println("Основне відхилення тета0 = " + teta0);**

**System.out.println();**

**double Fuv1 = Math.max(sigmaY1, sigmaY2) / Math.min(sigmaY1, sigmaY2);**

**double Fuv2 = Math.max(sigmaY1, sigmaY3) / Math.min(sigmaY1, sigmaY3);**

**double Fuv3 = Math.max(sigmaY3, sigmaY2) / Math.min(sigmaY3, sigmaY2);**

**System.out.println("Fuv1 = " + Fuv1);**

**System.out.println("Fuv2 = " + Fuv2);**

**System.out.println("Fuv3 = " + Fuv3);**

**System.out.println();**

**double tetauv1 = (numberOfY - 2.0) / (double)numberOfY \* Fuv1;**

**double tetauv2 = (numberOfY - 2.0) / (double)numberOfY \* Fuv2;**

**double tetauv3 = (numberOfY - 2.0) / (double)numberOfY \* Fuv3;**

**System.out.println("Тета uv1 = " + tetauv1);**

**System.out.println("Тета uv2 = " + tetauv2);**

**System.out.println("Тета uv3 = " + tetauv3);**

**System.out.println();**

**double Ruv1 = Math.abs(tetauv1 - 1) / teta0;**

**double Ruv2 = Math.abs(tetauv2 - 1) / teta0;**

**double Ruv3 = Math.abs(tetauv3 - 1) / teta0;**

**System.out.println("Ruv1 = " + Ruv1);**

**System.out.println("Ruv2 = " + Ruv2);**

**System.out.println("Ruv3 = " + Ruv3);**

**System.out.println();**

**if (Ruv1 < R\_critical && Ruv2 < R\_critical && Ruv3 < R\_critical) {**

**System.out.println("Ruv < Rkp, дисперсія однорідна!");**

**} else {**

**System.out.println("Ruv > Rkp, дисперсія неоднорідна!");**

**}**

**System.out.println();**

**}**

**private void normalize() {**

**double mx1 = (X11 + X12 + X13) / 3;**

**double mx2 = (X21 + X22 + X23) / 3;**

**double my = (averageY[0] + averageY[1] + averageY[2]) / 3;**

**double a1 = (X11\*X11 + X12\*X12 + X13\*X13) / 3;**

**double a2 = (X11\*X21 + X12\*X22 + X13\*X23) / 3;**

**double a3 = (X21\*X21 + X22\*X22 + X23\*X23) / 3;**

**double a11 = (X11\*averageY[0] + X12\*averageY[1] + X13\*averageY[2]) / 3;**

**double a22 = (X21\*averageY[0] + X22\*averageY[1] + X23\*averageY[2]) / 3;**

**double determinant = (a1\*a3+mx1\*a2\*mx2+mx2\*mx1\*a2 - mx2\*mx2\*a1-mx1\*mx1\*a3-a2\*a2);**

**b0 = (my\*a1\*a3+mx1\*a2\*a22+mx2\*a11\*a2-a22\*a1\*mx2-mx1\*a11\*a3-my\*a2\*a2)/determinant;**

**b1 = (a3\*a11+a22\*mx1\*mx2+a2\*my\*mx2 - mx2\*mx2\*a11-a22\*a2-mx1\*my\*a3)/determinant;**

**b2 = (a1\*a22+a2\*mx1\*my+mx1\*mx2\*a11 - mx2\*my\*a1-mx1\*mx1\*a22-a2\*a11)/determinant;**

**System.out.println("b0 = " + b0);**

**System.out.println("b1 = " + b1);**

**System.out.println("b2 = " + b2);**

**System.out.println();**

**System.out.println("Нормоване рівняння регресії: ");**

**System.out.println("y = (" + b0 + ") + (" + b1 + ") \* X1 + (" + b2 + ") \* X2");**

**System.out.println();**

**System.out.println("Перевірка:");**

**System.out.println((b0 + X11\*b1 + X21\*b2) + " = " + averageY[0]);**

**System.out.println((b0 + X12\*b1 + X22\*b2) + " = " + averageY[1]);**

**System.out.println((b0 + X13\*b1 + X23\*b2) + " = " + averageY[2]);**

**System.out.println();**

**}**

**private void naturalize() {**

**double deltaX1 = (maxX1 - minX1) / 2;**

**double deltaX2 = (maxX2 - minX2) / 2;**

**double X10 = (maxX1 + minX1) / 2;**

**double X20 = (maxX2 + minX2) / 2;**

**double a0 = b0 - b1\*X10/deltaX1 - b2\*X20/deltaX2;**

**double a1 = b1/deltaX1;**

**double a2 = b2/deltaX2;**

**System.out.println("a0 = " + a0);**

**System.out.println("a1 = " + a1);**

**System.out.println("a2 = " + a2);**

**System.out.println();**

**System.out.println("Натуралізоване рівняння регресії:");**

**System.out.println("y = (" + a0 + ") + (" + a1 + ") \* X1 + (" + a2 + ") \* X2");**

**System.out.println();**

**System.out.println("Перевірка:");**

**System.out.println((a0 + minX1\*a1 + minX2\*a2) + " = " + averageY[0]);**

**System.out.println((a0 + minX1\*a1 + maxX2\*a2) + " = " + averageY[1]);**

**System.out.println((a0 + maxX1\*a1 + minX2\*a2) + " = " + averageY[2]);**

**}**

**}**

**Main.java**

**package com;**

**/\*\***

**\* Created by User on 19.10.2016.**

**\*/**

**public class Main {**

**public static void main(String[] args) {**

**int minX1 = -20;**

**int maxX1 = 30;**

**int minX2 = 30;**

**int maxX2 = 80;**

**int minY = -830;**

**int maxY = -730;**

**int numberOfY = 6;**

**Experiment experiment = new Experiment(minX1, maxX1, minX2, maxX2, minY, maxY, numberOfY);**

**}**

**}**

**Висновки:**

У ході лабораторної роботи було досліджено двофакторний експеримент з лінійним рівнянням регресії, використано критерій Романовського для перевірки дисперсій на однорідність. Отримані результати при перевірці є адекватними.