Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Лабораторна робота №4

З предмету: «Теорія ймовірності»

біноміальний розподіл

Виконав: студент групи ІО-22

Щербина Микита

Київ 2013р.

**import** java.util.Scanner;

**public** **class** Main {

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

**int** countNumber = 1000;// длина последовательности

**int** m = 30;

**int**[] x = **new** **int**[countNumber];// генерируемая последовательность

**int** n = 1000;// количество проводимых испытаний

**double** p = 0.5;// вероятность успеха в одной испытании

BinomialDistribution bd = **new** BinomialDistribution(n, p);

**double** valueTest;

// автоматическое генерирование выборки

System.*out*.println("Значение параметров биномиального закона распределения");

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

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

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

x[i] = bd.nextNumber();

}

valueTest= ChiSquaredTest

.*chiSquaredValue*(countNumber, m, x, bd);

System.*out*.println("значение критерия Пирсона = " + valueTest);

System.*out*.println("вероятность правильности гипотезы = "

+ ChiSquaredTest.*valueProbability*(valueTest, m));

// ввод данных пользователем

m = 3;

countNumber = 10;

x = **new** **int**[countNumber];

Scanner in = **new** Scanner(System.*in*);

System.*out*.println();

System.*out*

.println("Введите параметры биномианального закона распределения");

System.*out*.print("n = ");

n = in.nextInt();

System.*out*.print("p = ");

p = in.nextDouble();

bd = **new** BinomialDistribution(n, p);

System.*out*.println("введите " + countNumber + " значений выборки");

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

x[i]=in.nextInt();

}

valueTest = ChiSquaredTest

.*chiSquaredValue*(countNumber, m, x, bd);

System.*out*.println("значение критерия Пирсона = " + valueTest);

System.*out*.println("вероятность правильности гипотезы = "

+ ChiSquaredTest.*valueProbability*(valueTest, m));

}

}

**public** **class** ChiSquaredTest {

**static** **final** **double** *maxValue*=10E10;

**public** **static** **double**[] *p* = { 1,0.99, 0.98, 0.95, 0.90, 0.80, 0.70, 0.50,

0.30, 0.20, 0.10, 0.05, 0.02, 0.01, 0.001,0 };

**public** **static** **double**[][] *valueChi* = {

// k=1

{ 0,0, 0.001, 0.004, 0.016, 0.064, 0.148, 0.455, 1.074, 1.642, 2.71,

3.84, 5.41, 6.64, 10.83,*maxValue* },

// k=27

{ 0,12.88, 14.12, 16.15, 18.11, 20.7, 22.7, 26.3, 30.3, 32.9, 36.7,

40.1, 44.1, 47.0, 55.5,*maxValue* } };

**static** **double** chiSquaredValue(**int** n, **int** m, **int**[] listValue,

BinomialDistribution b) {

Arrays.*sort*(listValue);

**double** step = (**double**) ((b.maxValue - b.minValue)) / m;

**double**[] segments = **new** **double**[m];

**double**[] listP = **new** **double**[m];

segments[0] = step;

listP[0] = b.getProbability(segments[0]);

**for** (**int** i = 1; i < segments.length; i++) {

segments[i] = segments[i - 1] + step;

listP[i] = b.getProbability(segments[i])

- b.getProbability(segments[i - 1]);

}

**int**[] listCountValue = **new** **int**[m];

**int** j = 0;

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

**if** (listValue[i] <= segments[j]) {

listCountValue[j]++;

} **else** {

j++;

listCountValue[j]++;

}

}

**double** result = 0;

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

**double** z = n \* listP[i];

**if** (z > 0.5)

result += Math.*pow*(listCountValue[i]-z, 2) / z;

}

**return** result;

}

**static** **double** valueProbability(**double** chiSquared, **int** m) {

**int** r = 2;

**int** k = m - r - 1;

**int** j = -1;

**if** (k == 0)

j = 0;

**if** (k == 27)

j = 1;

**double** a = *maxValue*;

**double** b = *maxValue*;

**double** p1 = 0;

**double** p2 = 0;

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

**if** (chiSquared < *valueChi*[j][i]) {

a = *valueChi*[j][i - 1];

b = *valueChi*[j][i];

p1 = *p*[i - 1];

p2 = *p*[i];

**break**;

}

**double** result = p1 - ((a - chiSquared) / (a - b)) \* (p1 - p2);

**return** result;

}

}

**Результати:**

Значение параметров биномиального закона распределения

n = 1000

p = 0.5

значение критерия Пирсона = 9.39005246499612

вероятность правильности гипотезы = 0.9927095865954999