Національний Технічний Університет України

«Київський Політехнічний Інститут»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Лабораторна робота №4

Перевірка гіпотез за критерієм Χ

|  |  |
| --- | --- |
| Прийняв  Доц. Марковський О.П.  «\_\_»\_\_\_\_\_\_\_\_ 2014 р. | Виконала Студентка 2-ого курсу ФІОТ  групи ІО-32  Руденко Т.А. |

Біноміальний розподіл

Лістинг

**import** java.util.Arrays;

**import** org.apache.commons.math3.distribution.BinomialDistribution;

**public** **class** Lab4 {

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

**int** l = 1000;

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

**int** n = 1000;

**double** p = 0.5;

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

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

x[i] = b.sample();

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

}

System.*out*.println();

System.*out*.println("Value of Pearson's chi-squared test: " + *xSquareTest*(x, 30, b));

System.*out*.println("Probability of trueness of the hypothese: " + *probability*(*xSquareTest*(x, 30, b), 30));

n = 5;

p = 0.4;

BinomialDistribution b1 = **new** BinomialDistribution(n, p);

**int** [] test = {2, 4, 1, 3, 3, 2, 2, 5, 2, 6};

System.*out*.println("Value of Pearson's chi-squared test: " + *xSquareTest*(test, 3, b1));

System.*out*.println("Probability of trueness of the hypothese: " + *probability*(*xSquareTest*(test, 3, b1), 3));

}

**public** **static** **double** xSquareTest(**int**[] sequence, **int** intNum, BinomialDistribution b){

Arrays.*sort*(sequence);

**int** [] bounds = **new** **int** [intNum];

**double** intLength = (**double**)(b.getSupportUpperBound()-b.getSupportLowerBound())/intNum;

bounds[0] = (**int**)Math.*ceil*(intLength);

**for**(**int** i = 1; i < intNum; i++){

bounds[i] = bounds[i-1]+(**int**)Math.*ceil*(intLength);

}

**int** [] valNum = **new** **int**[intNum];

**int** k = 0;

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

**if** (sequence[i] <= bounds[k]) {

valNum[k]++;

} **else** {

k++;

valNum[k]++;

}

}

**double** [] p = **new** **double**[intNum];

p[0] = b.cumulativeProbability(bounds[0]);

**for**(**int** i = 1; i < intNum; i++){

p[i] = b.cumulativeProbability(bounds[i])-b.cumulativeProbability(bounds[i-1]);

}

**double** xi = 0;

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

**double** z = sequence.length\*p[i];

**if** (z > 0.5)

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

}

**return** xi;

}

**public** **static** **double** probability(**double** xi, **int** intNum){

**double** prob = 0;

**double** xiApp = 0;

**double** [][] table = {{0, 0.01, 0.02, 0.04, 0.103, 0.211, 0.446, 0.713, 1.386, 2.41, 3.22, 4.6, 5.99, 7.82, 9.21, 13.82},

{0, 13.12, 14.26, 15.57, 17.71, 19.77, 22.5, 24.6, 28.3, 32.5, 35.1, 39.1, 42.6, 46.7, 49.6, 58.3}};

**double** [] probTable = {1, 0.995, 0.99, 0.98, 0.95, 0.9, 0.8, 0.7, 0.5, 0.3, 0.2, 0.1, 0.05, 0.02, 0.01, 0.001};

**if**(intNum == 30){

**for**(**int** i = 0; i < table[1].length-1; i++)

**if**(xi >= table[1][i] && xi <= table[1][i+1]){

xiApp = table[1][i+1];

prob = probTable[i+1];

}

}**else**

**for**(**int** i = 0; i < table[0].length-1; i++)

**if**(xi >= table[0][i] && xi <= table[0][i+1]){

xiApp = table[0][i+1];

prob = probTable[i+1];

}

**return** prob;

}

}