Лабораторна робота №5

Виконав: студент

ФІОТ групи ІО-43

Сербін Олесандр

Перевірив:

Марковський О. П.

Київ 2015

**Варіант: М, Д, 5 станів**

package labs;

import org.apache.commons.math3.linear.RealMatrix;

import org.apache.commons.math3.linear.\*;

import java.util.Random;

/\*\*

\* Created by Alexander Serbin on 12/8/2015.

\*/

public class lab5 {

public static int peaks=5;

public static double mor[][]={{0, 0.6, 0, 0, 0.4},

{0, 0.7, 0.3, 0, 0},

{0.5, 0.5, 0, 0,0},

{0, 0, 0.9, 0.1, 0},

{0.8, 0, 0, 0.2, 0}

};

public static double[][] m1;

public static double sum\_1=0, r;

public static double [] state=new double[peaks];

public static int p=1;

public static void main(String[] args) {

mdisplay(mor, "Original matrix:");

m1 =new double[mor.length][mor.length];

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

for (int j = 0; j< mor.length; j++){

sum\_1+= mor[i][j];

m1[i][j]=sum\_1;

}

sum\_1=0;

}

mdisplay(m1,"Modified matrix:");

for (int i=0;i<1000;i++){

Random ra=new Random();

r=ra.nextDouble();

for(int o=0;o<peaks;o++){

if (r <= m1[p][o]) {

state[p]++;

p = o;

break;

}

}

// System.out.println("r = " + r);

}

double[][] Matrix = new double[peaks][peaks];

double[] Vector = new double[peaks];

// Generating equations system

for (int i = 0; i < peaks; i++)

Matrix[0][i] = 1;

Vector[0] = 1;

for (int i = 1; i < peaks; i++) {

for (int j = 0; j < peaks; j++)

if (j == i)

for (int p = 0; p < peaks; p++) {

if (p != j)

Matrix[i][j] -= mor[j][p];

}

else

Matrix[i][j] = mor[j][i];

Vector[i] = 0;

}

RealMatrix coefficients = new Array2DRowRealMatrix(Matrix, false);

DecompositionSolver solver = new LUDecomposition(coefficients)

.getSolver();

RealVector constants = new ArrayRealVector(Vector, false);

RealVector solution = solver.solve(constants);

System.out.println();

System.out.println("Solution of linear equations:");

System.out.println();

for (int i = 0; i < peaks; i++)

System.out.println("P" + (i+1) + " = " + solution.getEntry(i));

System.out.println();

System.out.println("Amount of staying in every state:\n");

double s2=0;

for(int ii=0;ii<state.length;ii++){

double asd=(state[ii]/1000);

s2+=asd;

System.out.print("State no."+(ii+1)+": "+asd+"\n");

}

System.out.println(s2);

}

public static void mdisplay(double[][] m, String mname){

System.out.println();

System.out.println(mname+"\n");

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

for(int j=0; j<m.length;j++){

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

}

System.out.println();

}

}

}

**Результат:**

Original matrix:

0.0 0.6 0.0 0.0 0.4

0.0 0.7 0.3 0.0 0.0

0.5 0.5 0.0 0.0 0.0

0.0 0.0 0.9 0.1 0.0

0.8 0.0 0.0 0.2 0.0

Modified matrix:

0.0 0.6 0.6 0.6 1.0

0.0 0.7 1.0 1.0 1.0

0.5 1.0 1.0 1.0 1.0

0.0 0.0 0.9 1.0 1.0

0.8 0.8 0.8 1.0 1.0

Solution of linear equations:

P1 = 0.14053716427232987

P2 = 0.5996252342286071

P3 = 0.19113054341036853

P4 = 0.012492192379762644

P5 = 0.056214865708931895

Amount of staying in every state:

State no.1: 0.14

State no.2: 0.575

State no.3: 0.207

State no.4: 0.015

State no.5: 0.063