Національний технічний університет України «Київський політехнічний інститут»

Кафедра обчислювальної техніки

Лабораторна робота №3

з дісципліни **«Теорія ймовірності»**

Виконала:

студентка 2 курсу

ФІОТ гр. ІО-92

ГлуШко Ольга



Київ 2010 р.

package lab3;

public class Test {

public static void main(String[] args) {

int Number=1000;

double[] X=new double[Number];

double[] Y=new double[Number];

double mathExpectationX,standartDeviationX;

double mathExpectationRealX=0.0,mathExpectationRealY=0.0;

double sigmaRealX,sigmaRealY;

double lambda;

double corellationTeor,corellationReal;

mathExpectationX=0.0;//mathematical expectation for Normal

standartDeviationX=1.0;//sigma for Normal

lambda=2.0;//Lambda Exponential

corellationTeor=-0.0;

final double sqrt2 = Math.sqrt(2);

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

double Yj = 0;

for (int h = 0; h<=5; h++){

Yj += Math.random();

}

Yj = sqrt2 \* (Yj - 3);

double R = standartDeviationX\*Yj + mathExpectationX;

X[i]=R;

}

//exponential generator

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

Y[i]=(-1/lambda)\*Math.log(Math.random());

// System.out.println(Y[i]);

}

if(corellationTeor>0){

bubbleSort(X);

pozitiveCorellationSort(Y,X,corellationTeor);}

if(corellationTeor<0){

bubbleSort(X);

negativeCorellationSort(Y,X,corellationTeor);

}

corellationReal=cor(X,Y);

System.out.println("theoretical corelation = "+corellationTeor);

System.out.println("real corelation = "+corellationReal);

mathExpectationRealX=MathExp(X);

System.out.println("real math expectation for Normal=="+mathExpectationRealX);

sigmaRealX=StDev(X);

System.out.println("real sigma for Normal = "+sigmaRealX);

mathExpectationRealY=MathExp(Y);

System.out.println("real math expectation for Exponential = "+mathExpectationRealY);

sigmaRealY=StDev(Y);

System.out.println("real sigma for Exponential= "+sigmaRealY);

}

public static double MathExp(double[] mas){

double M\_mas=0.0;

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

M\_mas+=mas[i];

}

M\_mas/=mas.length;

return M\_mas;//real math expectation for array

}

public static double StDev(double[] mas){//standart deviation

double D\_mas=0.0,S\_mas=0.0;

double M\_mas=MathExp(mas);

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

D\_mas+=Math.pow(mas[i]-M\_mas,2);//pow - stepen'

}

D\_mas/=mas.length;

S\_mas=Math.sqrt(D\_mas);

return S\_mas;

}

public static double cov(double[] X,double[] Y){//covariation

double cov=0.0;

double MXt=MathExp(X);

double MYt=MathExp(Y);

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

cov+=(X[i]-MXt)\*(Y[i]-MYt);

}

cov/=X.length;

return cov;

}

public static double cor(double[] X,double[] Y){//corellation

double cor=0.0;

double SXt=StDev(X);

double SYt=StDev(Y);

double cov=cov(X,Y);

cor=cov/(SXt\*SYt);

return cor;

}

public static void bubbleSort(double[] mas){

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

for (int j=i;j<mas.length-1;j++){

if (mas[j+1]<mas[j]){

double buf=mas[j+1];

mas[j+1]=mas[j];

mas[j]=buf;

}

}

}

}

public static void pozitiveCorellationSort(double[] mas, double[] X, double corellationTeor){

double corellationReal=0; //unsorted Y,sorted X,corellationTeor

int length=mas.length;

for(int i=0;i<length;i++){//external Bubble sort

corellationReal=cor(X,mas);

corellationReal\*=100;

corellationReal=Math.round(corellationReal);

corellationReal/=100;

if(corellationReal==corellationTeor){

// System.out.println("number="+numb);

break;

}

for (int j=length-1;j>i;j--){//inner Bubble sort

if (mas[j]<mas[j-1]){

double buf=mas[j];

mas[j]=mas[j-1];

mas[j-1]=buf;

}

}

// numb++;

}

}

public static void negativeCorellationSort(double[] mas, double[] X, double corellationTeor){

double corellationReal=0; //unsorted Y,sorted X,corellationTeor

int length=mas.length;

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

corellationReal=cor(X,mas);

corellationReal\*=100;

corellationReal=Math.round(corellationReal);

corellationReal/=100;

if(corellationReal==corellationTeor){

break;

}

for (int j=0;j<length-1;j++){//inner Bubble sort

if (mas[j+1]>mas[j]){

double buf=mas[j+1];

mas[j+1]=mas[j];

mas[j]=buf;

}

}

}

}

}